Ano II - Nº 24 - Outubro / 1985 Cr$ 7.800

Progresso e Desenvolvimento na Informática Brasileira

TK 90X e TK 85
O diálogo possível

Para Manaus, Boa Vista, Porto Velho, Rio Branco, Altamira, Macapá, Rondônia (Via Aérea): Cr$ 10.140

Apresentamos o TK 2000 II. Ele roda o programa mais famoso do mundo.

De hoje em diante nenhuma empresa, por menor que seja, pode dispensar o TK 2000 II. Por que?

O novo TK 2000 II roda o Multicalc: a versão Microsoft do Visicalc®, o programa mais famoso em todo o mundo.

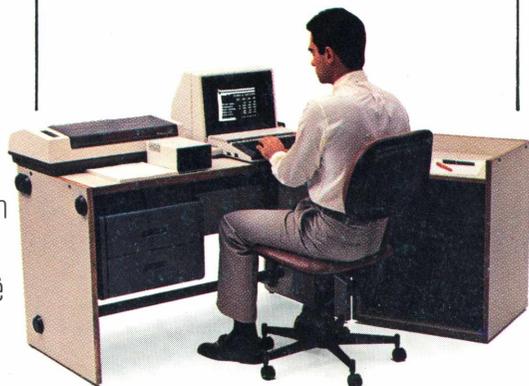
Isto significa que, com ele, você controla estoques, custos, contas a pagar, faz sua programação financeira, efetua a folha de pagamentos e administra minuto a minuto as suas atividades.

Detalhe importante: o novo TK 2000 II, com Multicalc, pode intercambiar planilhas com computadores da linha Apple®.

E, como todo business computer que se preza, ele tem teclado profissional, aceita monitor, diskette, impressora e já vem com interface.

Além de poder ser ligado ao seu televisor (cores ou P&B), oferecendo som e imagem da melhor qualidade.

Portanto, peça logo uma demonstração do novo TK 2000 II, nas versões 64K ou 128K de memória.

A mais nova estrela do show business só espera por isto para estrear no seu negócio.

Preço (128 K):
Cr$ 2.949.850

MICRODIGITAL
computadores pessoais

Open for Business.

® Marca registrada da Apple Computer.

Filiada à ABICOMP

® Marca registrada da Visicorp.

* Sujeito a alteração sem prévio aviso.

ÍNDICE

Capa: Hector Gomez Alísio

EXPEDIENTE

DIRETOR RESPONSÁVEL
Abraham Poppovich

PRODUÇÃO EDITORIAL
Álvaro A. L. Domingues

EDITORA
Ana Lúcia de Alcântara (M.T. 14495)

REDAÇÃO
Fábio Augusto Polônio
Marcos Lorenzi
Tânia M. Cristina Batista (Secretária)
Mônica Rocha (Redatora)

ASSESSORIA TÉCNICA
Gustavo Egídio de Almeida
Paulo Lauand
Wilson José Tucci

CORRESPONDENTES
Fátima França — Rio de Janeiro

PROGRAMAÇÃO VISUAL
Walter de Jesus

COLABORADORES
César de Afonseca e Silva Neto, Wilson José Tucci, Christiano A.C. Nasser, Renato da Silva Oliveira, Gustavo Egídio de Almeida, Victor Meyer, Ralph Marques Aguiar, Solange Aparecida Menezes (revisão)

MARKETING
Aurio José Mosolino (supervisor)
Eduardo Garcia Souza

ASSINATURAS
Marli Mantovani

CIRCULAÇÃO
José Aparecido Bueno

ADMINISTRAÇÃO
Cleusa Ap. S. Malian

DISTRIBUIÇÃO
Fernando Chinaglia Distribuidora S/A.

DIAGRAMAÇÃO, ARTE, FOTOCOMPOSIÇÃO, FOTOLITO E IMPRESSÃO
Bandeirante S/A. Gráfica e Editora.

MICROHOBBY é editada mensalmente por Micromega Publicações e Material Didático Ltda.
Endereço para Correspondência:
Av. Angélica, 2318 — 14.º andar
Cx. Postal 54096 — CEP 01295
São Paulo — SP — Fone: (011) 255-0366.

Para solicitar assinatura anual utilize o encarte nesta Revista e pague em qualquer agência do Banco Bradesco.

MICROHOBBY 24

OUTUBRO/85

EDITORIAL

A informática brasileira mostrou,durante a realização dos seus dois maiores eventos, um real amadurecimento, seja por parte do material apresentado na Feira, e pelos temas discutidos no Congresso ou pelo teor político que revestiu todo o Evento.

Este ano não houve a parafernália de inúmeros lançamentos na linha dos micropessoais, mas sim uma procura grande de informações, por parte dos usuários que buscaram,principalmente, o que as empresas estavam apresentando em softwares e periféricos.

Uma das tendências da Feira foram os lançamentos de micros de linhas mais superiores, voltados a aplicações profissionais.

Equipamentos de 8 bits e de 16 bits registraram a tendência da próxima temporada.Ligados entre si, formando redes, esses computadores demonstraram que a informática no Brasil já superou a sua fase inicial, graças a própria evolução do uso que os usuários fazem de seus equipamentos.

Nesta edição dedicamos a Microhobby para este grande evento da área, que mereceu, este ano a atenção de vários setores da sociedade civil.

O "Micropress" foi todo ele voltado à cobertura do Congresso e da Feira de Informática. Apresentamos os lançamentos, em periféricos e softwares; a tecnologia nacional mostrada pelas universidades,e os melhores momentos das palestras e debates.

As seções habituais não foram esquecidas e acredito que os outros artigos deste número deixarão vocês bem satisfeitos.

Ana Lúcia de Alcântara

Se você possui um TK 2000, de hoje em diante não pode mais dispensar os programas Microidéia para o seu micro.

Com eles, você controla estoques, custos, receitas e contas bancárias. Programa as finanças domésticas e as de suas empresas. E cadastra seus clientes, fornecedores ou amigos.

O software Microidéia vai transformar seu TK 2000 numa poderosa ferramenta profissional, pessoal ou doméstica. Capaz de realizar em segundos tarefas que lhe tomavam um grande tempo e esforço.

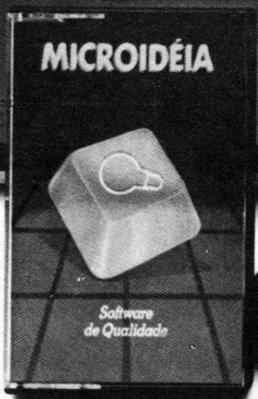
Totalmente desenvolvidos no Brasil, todas as instruções de tela e manuais são em português. E toda vez que lançarmos uma nova versão de um software, você poderá trocá-la por seu programa original.

Em cassete ou diskette, já temos para TK 2000: Orçamento Doméstico, Controle Bancário, Mala Direta, Mini Banco de Dados, Fluxo de Caixa, Contas a Pagar, Contas a Receber e Controle de Estoques, todos compatíveis com o Apple e TK 2000 II.

Procure já um dos nossos revendedores e abra seu TK 2000 para o software Microidéia. E sinta uma nova estrela nascendo ao seu lado.

Mas, se em sua cidade não tem revendedor Microidéia, peça qualquer um destes programas pelo correio.

Em cassete eles custam Cr$ 55 mil. Em diskette, o preço é Cr$ 250 mil.

TK 2000. Open for Software Microidéia.

Sim!
Quero adquirir o(s) seguinte(s) programa(s) Microidéia:

Programa	K7	Disk	Quant.	Preço
TOTAL				

Estou enviando um cheque do banco ____________ de n.º ____________ nominal à Microidéia Ltda.

Microidéia
Caixa Postal 6151
20022 / Rio de Janeiro / RJ

☐ **Envie-me um catálogo de programas Microidéia.**

Nome ____________
End. ____________
Bairro ____________ Cidade ____________
Tel.: ____________ CEP ____________
Micro ____________

Prezado Senhor,

Lendo sua revista, vi que vocês querem que os leitores lhes enviêm sugestões; a minha é de que vocês escrevam mais artigos sobre o TK 2000 COLOR, fazendo programas de jogos, e indicando software disponíveis, e editando também os seus respectivos preços.

Seria bom também que editassem programas em Assembly.

Alex Settini Sohler.
Valença - RJ.

Caro Alex,

Cartas sugestivas como a sua é que fazem com que a MICROHOBBY atenda da melhor forma possível, os seus leitores, melhorando dessa maneira, a nossa linha editorial. Suas sugestões são muito válidas. É através das opiniões de nossos leitores que a Revista evolui. Anotamos todas as suas dicas em nosso arquivo de pautas e tão logo seja possível as incluiremos nos temas de posteriores edições.

Prezado Senhor,

A dúvida é a respeito do modo de edição do TK-2000. Como devemos proceder após ter dado entrada numa linha de um programa, se observarmos nesta linha algum erro?

Paulo Florêncio S. de Menezes.
Manaus - AM.

Caro Paulo,

Se você não possui o programa "EDITOR" desenvolvido pela Microsoft o único recurso que lhe resta é a redigitação de toda a linha.

O "Editor" éum programa em Linguagem de Máquina, que foi desenvolvido com o objetivo de auxiliar os usuários em seus trabalhos de edição em programação Basic, para o TK-2000. Para obter maiores informações ligue para o Setor de Apoio ao Usuário da Microdigital Eletrônica, no telefone: 255-0366/r.61 e fale com o Dirceu.

Prezado Senhor,

Elaborei um programa totalmente em BASIC, muito longo, que mais ou menos no meio do programa, cada linha que se dava entrada, o vídeo se alterava e, quando listei o programa só havia 1/3 do que tinha digitado. Consultando o manual, descobri o meu erro, deveria digitar MP para liberar a memória e assim o fiz. Salvei o programa em fita K-7 e agora este só entra digitando-se MP antes de LOAD T. Como fazer para que este sistema seja automático, como nas fitas já gravadas?

2 - Há um meio de "entrar" num programa bloqueado? Como?

Pedro Antonio Ribeiro
São Paulo - São Paulo.

Caro Pedro,

Para acionar o comando MP, automaticamente, durante o carregameno de um programa, é preciso criar uma rotina em Linguagem de Máquina. Além disso,há necessidade de descobrir qual o endereço correspondente a este comando para que a partir deste, você possa fazer MP ser acionado automaticamente.

É comum os usuários encontrarem programas que estejam bloqueados, mas isto não significa que seja impossível penetrá-los. Porém é nenecessário ter um bom conhecimento de Linguagem de Máquina, para poder descobrir qual o endereço inicial da rotina, usada para bloqueá-lo.

Prezado Senhor,

Esta tem a finalidade de parabenizá-los pelo excelente nível que a revista vem alcançando.

Gostaria de estender as congratulações ao Arold P. Carvalho pelo seu magnifico artigo - "Explorando a alta-resolução TK-2000". É um programa de grande utilidade e fácil manuseio.

Gostaria, no entanto, de avisar que o dito programa contém dois erros de digitação: o primeiro é tão evidente que não passa desapercebido a ninguém: na linha 855, em vez de "OR" esta "IR".

O segundo erro é mais difícil de achar, pois com o ONERR GOTO ativado, não acusa erro de Sintaxe. O erro está na linha 930 - está faltando dois pontos (:) entre O";" e o HTAB 36.

Estes dados referem-se a rotina da cópia do já citado programa, e está na MICROHOBBY 21; pág. 52.

Esperamos que novos programas excelentes como este continuem a ser apresentados.

J.J. de Araujo Moura Filho
Rio de Janeiro - RJ.

Caro J.J. de Araujo,

Agradecemos seus votos de parabéns. Isto nos faz ter a certeza de que estamos no caminho certo. Suas congratulações nos tornam realizados e nos alertam, ao mesmo tempo, com referência as nossas falhas.

Prezado Senhor,

Gostaria de pedir-lhes o obséquio de indicar-me a maneira de adaptar o jogo "Base Lunar" publicado na MICROHOBBY nro. 21 para o uso do "joystick" pois tal como está os movimentos da mira na tela não correspondem aos movimentos feitos com o "joystick". Aliás, parabéns ao Sr. Fábio Polonio pelo excelente visual do jogo.

Theodomiro Aguiar
Petrópolis - RJ

Caro Theodomiro,

Nosso programador agradece seus elogios. Quanto a informação solicitada, a respeito do Joystick no TK 90X, proceda da seguinte maneira: na linha 950 altere os números que foram definidos para o teclado.

Para a opção de uso do joystick, utilize os números 6, 7, 8, 9 e 0. Os números 6 e 7 correspondem aos movimentos laterais para a esquerda e direita, respectivamente. Os números 8 e 9 são para os movimentos verticais, de descida (8) e subida (9). Para atirar, utilize o número 0.

Prezado Senhor,

Sou assinante da MICROHOBBY desde praticamente o início e não tenho críticas a fazer, muito pelo contrário sou apreciador e já fiz vários amigos assinarem também.Posso fazer um pedido? Por razões óbvias solicito mais intruções sobre o TK 2000 e como adaptar programas para Apple para o TK.

Muito grato. Parabéns e votos de muito sucesso.

Antonio Alberto P. da Silva
Belo Horizonte - MG

Caro Antonio,

Foi com grande satisfação que a equipe da MICROHOBBY recebeu sua carta. Nosso principal objetivo não é apenas manter nossos leitores bem informados, mas também fornecer o suporte necessário para que possam utilizar seus equipamentos em toda totalidade.

Para isto, tentamos solucionar as dúvidas da melhor forma possível. Quando não está ao nosso alcance procuramos nos informar em outras fontes para respondê-las.

São cartas como a sua que tornam nosso trabalho ainda mais credibilizado.

Quanto ao seu pedido sobre adaptação do Apple ao TK-2000, solicito-lhe que se comunique com o Setor de Apoio ao usuário do fabricante deste equipamento.

Clube de Usuários

TK-82C/83/85

Diarone Broilo
Caixa Postal 324
95100 - Caxias do Sul - RS

Eduardo Saita
Rua Ernesto Onizzolo 157
13290 - Louveira - SP

Fernando Antonio Silva
R. Noronha Torrezão 141 - ap. 304
24240 - Niteroi - RJ

H.S. Ferreira
Caixa Postal 81825
27500 - Resende - RJ

José Afonso P. Ferreira Terttius
Shin QI 4 Cj.1 casa 21 - Lago Norte
71500 - Brasilia - DF

José Carlos Rodrigues
R. Antonio Bento 226 - ap. 24
11100 - Santos - SP

TK-2000/2000II

Alexandre Lauria Ribeiro
R. Bolivia 26
12280 - Caçapava - SP

Alexandre Maia Cardoso
R. Guimarães Rosa 143/I607 - Barra Tijuca
22700 - Rio de Janeiro - RJ

Altair Mario Wielewski
Caixa Postal 31
87530 - Xambre - PR

Carlos Alberto Rocha Sobrinho
R. Subtte. Francisco Hiero 165
80000 - Curitiba - PR

Claudio N. de Carvalho
R. Paulino Teixeira 2235/703
90000 - Porto Alegre - RS

Edy Belmonte Kich
R. Honorato Toniolo 76
90000 - Porto Alegre - RS

Observação

Para ter seu nome publicado, envie correspondência para:Microhobby, Clube do Usuário, Caixa Postal 54096, São Paulo - SP - CEP 01296. Indique o computador que possui, as interfaces e a sua área de interesse. Nosso clube está aberto a usuários de: TK-82/83, TK-85 e compatíveis, TK2000, Apple e compatíveis. É bastante útil a sua opinião e sugestão para melhorarmos esta seção. Escreva-nos também sobre suas experiências e de seus amigos. Estamos aguardando.

Rancho

Marcos Lorenzi

Este programa educativo, desenvolvido pela Multisoft Informática Ltda., destina-se ao micro pessoal da linha TK 90X.

Tentando familiarizar a criança desde pequena com o micro, a empresa criou um programa voltado para o público infantil, que permite à criança desenvolver sua imaginação e criatividade, manipulando figuras animadas.

Apesar de ser um programa educativo,"Rancho" ao mesmo tempo que ensina, permite que a criança brinque, evitando dessa forma a monotomia do software.

Características Gerais do Programa

O programa é composto de quatro tipos de telas:

1 - tela de composição;
2 - telas dos Elementos (no caso são duas);
3 - tela de Utilidade.

Para se manejar o programa utiliza-se um único cursor, que é comandado por cinco tipos de teclas. Este ainda possui um menu de comandos para as composições com seis opções.

Funcionamento do Rancho

Terminada a carga do programa, efetua-se a montagem e animação da tela de demonstração. Para iniciar o jogo pressione a tecla SPACE. Você pode manipular o cursor para os comandos que desejar, utilizando as teclas de manipulação do mesmo.

O usuário poderá animar o jogo com música, controlando a melodia; reproduzir elementos na tela de composição, quantas vezes desejar e apagar as figuras que se apresentam na tela.

Caso o usuário queira alterar as cores, poderá fazê-lo de acordo com o seu estilo, ou até mesmo mudar as figuras de posição, modificando o aspecto inicial da tela. A animação das personagens é feita individualmente, bastando apenas ir selecionando as figuras que serão animadas. Você poderá fazer os elementos viajarem, mas é possível fazê-lo somente com uma figura de cada vez.

O programa permite que se introduza 12 figuras de uma só vez, bastando apenas selecionar as figuras na tela dos elementos, e trazê-los para a tela de composição.

Para gravar ou ler o que foi realizado, basta seguir as intruções que estão no manual que acompanha o programa, onde as explicações estão muito bem detalhadas e são de fácil entendimento.

Comentários Finais

Por ser um programa educativo voltado ao pessoal infantil, não apresenta dificuldades de entendimento.

Com alguns contatos com o material e tendo alguém para dar as dicas de como utilizá-lo o jovem usuário estará apto a criar e desenvolver qualquer tipo de animação, que bem desejar. Além do mais, ele estará também se familiarizando com o micro. Em breve este software estará sendo comercializado no mercado, e poderá ser encontrado nos revendedores Fotóptica.

Decathlon

Marcos Lorenzi

Participe dos jogos olímpicos utilizando o 90X.

Você que sempre desejou participar de uma Olímpiada para poder mostrar suas qualidades como atleta, mas nunca teve uma oportunidade, agora, sem fazer muito esforço e utilizando um micro pessoal, você poderá reunir seus amigos e demonstrar-lhes o que é preciso para se tornar um verdadeiro atleta. Destinado ao TK 90X, Decathlon foi desenvolvido pela Multisoft e será comercializado em fita cassete, acompanhado de um manual de instruções.

Características gerais do programa

Após o processo habitual de carga do programa,surgirá na tela um menu de opções de controle.

Você deverá selecionar 1 ou 4 (respectivamente teclado e joystick). Se a opção foi 1 (teclado), então o usuário deverá determinar se as teclas de comando serão modificadas ou não, ficando ao seu critério a opção. Em seguida, por meio de teclado ou joystick, você introduzirá suas iniciais para que constem no placar caso seja classificado.

Voce participará de cinco provas do Decatlon (modalidade esportiva composta por dez provas), nas quais seu objetivo será a obtenção do menor tempo e o maior número de pontos. O tempo mínimo exigido (mostrado no placar) para classificação é regressivo, o que torna a prova mais dificil à cada etapa.

O placar exibe suas iniciais, o resultado obtido nas tentativas, os pontos acumulados e o tempo mínimo necessário para a classificação.

O jogo apresenta uma ótima resolusão, preocupando-se com os mínimos detalhes, para tornar o jogo o mais realístico possível.Os efeitos sonoros, muito bem definidos para cada situação, transmitem a você a sensação de estar realmente num estádio olímpico.

Comentários finais

Sua atenção estará voltada para um único ponto, o seu vídeo, onde sua preocupação será baixar os tempos das provas e superar os recordes já existentes.

Este jogo atrairá tanto o público jovem como o de meia idade, mas só vencerá aquele que tiver maior fôlego e melhor preparo físico (pelo menos nos dedos...).

Alta-Resolução para TK 85

Visando explorar ainda mais a capacidade que este equipamento possui em gráficos, resolvemos publicar um artigo que permita introduzir alta-resolução no TK 85.

Este programa abre espaço para que o usuário possa elaborar programas em alta-resolução, ou até mesmo carregar programas pré-elaborados que tenham esta característica. Este é armazenado na RAMTOP do equipamento, incluindo a tela de alta-resolução e o conjunto de caracteres (UDG), ocupando 7000 bytes de memória.

Como o usuário deve operar o carregador Hexadecimal

Em primeiro lugar o usuário deverá criar uma linha REM (97caracteres) e editá-la até que obtenha linhas REM de 1 a 7. Em seguida, introduzir POKEs na sequência mostrada na tabela 1 como comando direto.

Para verificar se as linhas REM estão com o tamanho correto, digite a seguinte sentença:

PRINT PEEK 16396 + 256 * PEEK 16397

Se surgir na tela o seguinte resultado: 17230, significa que está correto. Então ele deve digitar o programa apresentado na tabela 2. Após a digitação deve-se rodar o programa. Surgirá em sua tela uma linha " 0 REM", mas o programa não aparecerá no vídeo. Então, deve-se "deletar" as três linhas do programa que o usuário digitou e entrar com o programa "Carregador Hexadecimal".

Durante a digitação do carregador hexadecimal, só aparecerá a linha "0 REM". No caso do usuário se perder na digitação do programa, a melhor saída é editar a última linha digitada.

Ao final da digitação ele deve executar o programa, que pedirá ao usuário que entre com o endereço inicial e final e então permitirá iniciar-se a digitação dos códigos em hexadecimal que estão expostos na tabela 3. Ao final da digitação dos códigos hexadecimais, o usuário irá "deletar" o programa "Carregador Hexadecimal",linha por linha, para em seguida entrar direto com o programa BASIC (listagem 2).

Em seguida, entrar com o programa BASIC (listagem 2). Para salvá-lo dê um "auto-RUN", utilizando "GOTO 350".

Trabalhando com a Alta-Resolução

O programa, que contém os códigos,transforma a tela normal de texto em uma tela de alta-resolução e o código de um caractere qualquer no modo normal de tela, correspondem a oito códigos do UDG.

Ao rodar o programa BASIC, o usuário deverá responder se deseja definir o UDG ou não. Se a resposta for negativa, o programa será armazenado na RAMTOP, incluindo-se o conjunto de caracteres que permanecerá vazio. O conjunto de caracteres se inicia no endereço 16717, e na RAMTOP no 25772.

Se a resposta for positiva, então o computador perguntará por um código de qualquer caractere, ou seja, um número entre (2-63). Em seguida, ele pedirá que o usuário entre com oito números para construir o UDG. Completada a definição do UDG, ele terá duas opções para escolha: na primeira ele poderá salvar o conjunto de caracteres, incluindo o programa BASIC completo, em fita ou não. Caso a opção seja negativa, estas serão automaticamente armazenadas na RAMTOP.

Quando o programa BASIC estiver armazenado na RAMTOP, o usuário poderá criar seu próprio programa ou carregar um pré-existente. Nos dois casos citados, o programa deverá iniciar-se em RAND USR 25614, pois esta instrução colocará o computador no modo de alta-resolução e, na finalização do mesmo, deverá ser usado RAND USR 25602, passando o equipamento para o modo normal. O usuário poderá transformar a tela em altaresolução, usando como comando direto RAND USR 25686 e RAND USR 25626 para limpar a tela quando esta estiver no modo de alta-resolução.

Para remover um caractere da tela no modo de alta-resolução, o usuário deverá usar PRINT CHR $ (1), pois a rotina de transformação executa apenas remoção de CHR $ (1) e não CHR$(0) da tela no modo normal.

O usuário não pode esquecer de que nunca deverá usar o comando SCROLL, porque este poderá causar um "Crash" no equipamento.

```
POKE 16510,0
POKE 16511,205
POKE 16512,2
POKE 16514,118
POKE 16515,118
```

Tabela 1

```
10 FOR A=16516 TO 17228
20 IF PEEK A<>158 THEN POKE A,
158
30 NEXT A
```

Tabela 2

```
16618 - 9202CD2002DD212D - 686
16626 - 64C3A402E92A0C40 - 812
16634 - 2311006706187EFE - 565
16642 - 00280BFE01200236 - 394
16650 - 00C5CD8464C1237E - 988
16658 - FE76200C23EBC501 - 884
16666 - E90009EBC110DFC9 - 1110
16674 - 1318DBE5D521AC64 - 1009
16682 - E63F06004FCB21CB - 817
16690 - 10CB21CB10CB21CB - 910
16698 - 1009EB06081A7713 - 438
16706 - D51121001 9D110F5 - 758
16714 - D1E1C9 - 635
16514 - 7676210064220440 - 471
16522 - 21A14011006401B5 - 557
16530 - 02EDB0CDC3030000 - 818
16538 - 00760002AE02EA76 - 648
16546 - 76CD2A0A3E1EED47 - 775
16554 - DD218102C9CD1A64 - 917
16562 - 3E08ED47DD212D64 - 777
16570 - C92100670EC00620 - 581
16578 - 369E2310FB36C923 - 804
16586 - 0D20F3C921DFE611 - 992
16594 - 2100F30EFE061610 - 588
16602 - FE06C0ED78D3FF19 - 1300
16610 - CD556405C23C64CD - 954
```

Tabela 3

```
  99 REM CARREGADOR HEXADECIMAL
 100 PRINT "ENDERECO INICIAL"
 110 INPUT I
 120 PRINT "ENDERECO FINAL"
 130 INPUT F
 140 FOR N=I TO F STEP 8
 150 LET T=0
 160 PRINT N;" - ";
 170 INPUT A$
 180 PRINT A$;" - ";
 190 INPUT TOT
 200 PRINT TOT
 210 LET Z=0
 220 FOR K=1 TO LEN A$ STEP 2
 230 LET C=(CODE A$(K)-28)*16+CO
DE A$(K+1)-28
 240 LET T=T+C
 250 POKE N+Z,C
 260 LET Z=Z+1
 270 NEXT K
 280 IF TOT=T THEN GOTO 310
 290 PRINT "ERRO, ENTRE A NOVAME
NTE"
 300 GOTO 150
 310 NEXT N
```

Listagem 1

```
   5 REM        BASIC
   6 REM
  10 REM ALTA - RESOLUCAO
  20 PRINT "ENTRE COM OS CODIGOS
PARA FORMAR UDG""S?(S/N)"
  30 IF INKEY$="" THEN GOTO 30
  35 LET A$=INKEY$
  40 IF A$="N" THEN RAND USR 165
14
  50 IF A$="S" THEN GOTO 100
  60 GOTO 30
 100 CLS
 110 PRINT "NUMERO DO CARACTERE?
(2-63)"
 120 INPUT NC
 130 IF NC<2 OR NC>63 THEN GOTO
120
 140 PRINT NC
 150 PRINT "ENTRE COM OITO CODIG
OS..."
 160 LET X=16717+(NC*8)
 170 FOR A=X TO X+7
 180 INPUT C
 190 PRINT C
 200 POKE A,C
 210 NEXT A
 220 PRINT "MAIS CARACTERES?(S/N
)"
 230 IF INKEY$="" THEN GOTO 230
 240 LET A$=INKEY$
 250 IF A$="S" THEN GOTO 100
 260 IF A$<>"N" THEN GOTO 230
 265 IF INKEY$<>"" THEN GOTO 265
 270 PRINT "SALVAR OS CARACTERES
?(S/N)"
 280 IF INKEY$="" THEN GOTO 280
 290 LET A$=INKEY$
 300 IF A$="N" THEN RAND USR 165
14
 310 IF A$<>"S" THEN GOTO 280
 320 CLS
 325 IF INKEY$<>"" THEN GOTO 325
 330 PRINT "PRESSIONE QUALQUER T
ECLA PARA SALVAR..."
 340 IF INKEY$="" THEN GOTO 340
 345 CLS
 350 SAVE "ALTA-RESOLUCAO"
 360 RUN 20
```

Listagem 2

Periscópio

Victor Meyer

Victor Meyer é usuário do TK 85, leitor da Revista e participa desta edição com um programa, desenvolvido por ele, compatível com este microcomputador.

texto final: Mônica Rocha

O jogo possui dois obstáculos, que deverão ser ultrapassados pelo usuário. O primeiro consiste de um mapa/radar, através do qual o submarino deverá alcançar uma esquadra de navios inimigos, sem chocar-se com a moldura do radar e nem bater em qualquer barreira. Se acaso o jogador não conseguir efetuar esta prova, aparecerá uma mensagem na tela e automatica mente ocorrerá a sua desclassificação. No entanto, se ele obtiver sucesso ao atingir a esquadra, mas chocar-se com a moldura do radar, o jogo retornará ao início, fornecendo ao jogador três chances, ao fim das quais, ele perderá, se não aproveitá-las.

O obstáculo número dois tem como objetivo a visualização dos navios da esquadra inimiga. Eles deverão ser abatidos, através da tecla 0, com um total de dez tiros, sendo indicado ao final, o número de disparos e de navios afundados. Porém, enquanto o submarino estiver a tona, é necessário cuidado, pois um torpedo é lançado pela esquadra, em sua direção. Isto é indicado, por intermédio de uma numeração decrescente, tal como um radar de proteção que indica a aproximação de disparos. Desta forma, só é possível escapar submergindo o submarino, mas tendo um imenso cuidado para não ultrapassar a profundidade de dois asteriscos (indicados no vídeo), para não afundá-lo. Se isto acontecer, aparecerá uma mensagem no vídeo demonstrando a desclassificação do jogador.

O "Lança torpedos" pode ser movimentado na horizontal, em perseguição ao navio inimigo, até um certo ponto.

```
****************************

0-DISPARADOR
5-SUB<=
6-SUBMERGIR
7-EMERGIR
8-SUB=>
M-MAPA/RADAR
P-PERISCOPIO
```

Figura 1: Tela de apresentação

```
EM 1943 TODOS OS COMANDAN
TES DE SUBMARINO RECEBEM
UMA ORDEM EXPRESSA, CORTAR
A ROTA DE SUPRIMENTOS DOS
ALIADOS
```

Figura 2: Tela secundária

Figura 3: Primeira tela do jogo

Figura 4: Segunda tela do jogo

```
  1 LET W=4
  5 CLS
  7 LET W=W-1
  9 IF W=0 THEN GOTO 3095
 10 PRINT AT 1,2;"▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀";AT 2,2;"▌";AT 2
,3;"▄▄";AT 2,6;"▄▄";AT 2,9;"▄▄";
AT 2,12;"▖";AT 2,14;"▄▄";AT 2,17
;"▄▄";AT 2,20;"▄▄";AT 2,23;"▄▄";
AT 2,25;"▖";AT 2,27;"▄▄";AT 2,29
;"▌"
 20 PRINT AT 3,1;"█";AT 3,3;"▙"
;AT 3,4;"▟";AT 3,6;"▙";AT 3,7;"▖
";AT 3,9;"▙";AT 3,10;"▟";AT 3,12
;"▌";AT 3,14;"▙";AT 3,15;"▄";AT
3,17;"▌";AT 3,20;"▌";AT 3,21;"▌"
;AT 3,23;"▙";AT 3,24;"▟";AT 3,25
;"▌";AT 3,27;"▌";AT 3,28;"▌";AT
3,30;"█"
 30 PRINT AT 4,1;"▀";AT 4,2;"▌"
;AT 4,3;"▌";AT 4,6;"▙▄";AT 4,9;"
▌";AT 4,10;"▀";AT 4,11;"▖";AT 4,
12;"▌";AT 4,14;"▄▟";AT 4,17;"▙";
AT 4,18;"▄";AT 4,20;"▙";AT 4,21;
"▟";AT 4,23;"▌";AT 4,25;"▌";AT 4
,27;"▙";AT 4,28;"▟";AT 4,29;"▌";
AT 4,30;"▀"
 40 PRINT AT 5,2;"▙";AT 5,3;"▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄";AT 5,2
9;"▟"
 90 PRINT AT 12,2;"0-DISPARADOR
";AT 13,2;"5-SUB<=";AT 14,2;"6-S
UBMERGIR";AT 15,2;"7-EMERGIR";AT
 16,2;"8-SUB=>";AT 17,2;"M-MAPA/
RADAR";AT 18,2;"P-PERISCOPIO"
 100 FOR M=0 TO 15
 110 PRINT AT 8,2;"*************
***************"
 120 PRINT AT 8,2;"
                  "
 130 NEXT M
```

→

```
 140 PRINT AT 21,5;"DIGITE M OU
P"
 150 INPUT P$
 170 IF P$="M" THEN GOTO 175
 173 IF P$="P" THEN GOTO 1500
 175 CLS
 180 PRINT AT 1,9;"▌";AT 1,21;"▌
";AT 1,10;"▛";AT 1,22;"▛";AT 1,1
1;"▀";AT 1,23;"▀"
 190 PRINT AT 2,9;"▙";AT 2,10;"█
";AT 2,11;"▜";AT 2,14;"█";AT 2,1
6;"█";AT 2,18;"█";AT 2,21;"▙";AT
 2,22;"█";AT 2,23;"▜"
 200 PRINT AT 3,9;"▄";AT 3,21;"▄
";AT 3,10;"▟";AT 3,22;"▟";AT 3,1
1;"▐";AT 3,23;"▐"
 210 PRINT
 220 PRINT AT 5,4;"EM 1943 TODOS
 OS COMANDAN";AT 6,4;"TES DE SUB
MARINO RECEBEM ";AT 7,4;"UMA ORD
EM EXPRESSA, CORTAR";AT 8,4;"A R
OTA DE SUPRIMENTOS DOS"
 225 PRINT AT 9,4;"ALIADOS"
 230 PRINT AT 11,10;"█";AT 11,22
;"█"
 240 PRINT AT 12,16;"▄";AT 13,15
;"▄";AT 13,16;"█";AT 13,17;"▄";A
T 14,5;"▄▄▄▄▄▄▄▄▄▄";AT 14,20;"▄▄▄
▄";AT 14,15;"█";AT 15,15;"█";AT
14,16;"█";AT 14,17;"█";AT 14,18;
"█";AT 14,19;"▄"
 250 PRINT AT 15,6;"▀";AT 15,7;"
▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";AT 15,25;"▀
"
 260 PRINT AT 16,8;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒"
 270 FOR L=0 TO 21
 290 NEXT L
 295 FOR M=0 TO 90
 300 NEXT M
 490 CLS
 500 RAND
 510 FAST
 520 FOR E=0 TO 31
 530 PRINT AT 0,E;"▄";AT 21,E;"▀
"
 540 NEXT E
 550 FOR F=1 TO 20
 553 PRINT AT F,0;"▌";AT F,31;"▌
"
 570 NEXT F
 580 SLOW
 590 LET B=28
 600 LET C=10
 610 LET X=20
 620 LET Y=2
 630 FOR D=0 TO 500
 640 LET X=X+(2 AND INKEY$="6")-
(2 AND INKEY$="7")
 645 LET Y=Y+(2 AND INKEY$="8")-
(2 AND INKEY$="5")
 647 IF Y=(18) AND X=(20) OR Y=(
18) AND X=(19) OR Y=(18) AND X=(
18) OR Y=(18) AND X=(17) OR Y=(1
8) AND X=(16) THEN GOTO 5
 650 IF Y=(18) AND X=(1) OR Y=(1
8) AND X=(2) OR Y=(18) AND X=(3)
 OR Y=(18) AND X=(4) OR Y=(18) A
ND X=(5) THEN GOTO 5
 655 IF Y=(20) AND X=(7) OR Y=(2
0) AND X=(8) OR Y=(20) AND X=(9)
 OR Y=(20) AND X=(10) OR Y=(20)
AND X=(11) OR Y=(20) AND X=(12)
OR Y=(20) AND X=(13) OR Y=(20) A
ND X=(14) THEN GOTO 5
 668 PRINT AT 0,11;"MAPA/RADAR"
 670 PRINT AT 7,20;"▌";AT 8,20;"
▌";AT 9,20;"▌";AT 10,20;"▌";AT 1
1,20;AT 12,20;"▌";AT 13,20;"▌";A
T 14,20;"▌"
 675 PRINT AT 1,18;"▌";AT 2,18;"
▌";AT 3,18;"▌";AT 4,18;"▌";AT 5,
18;"▌";AT 20,18;"▌";AT 19,18;"▌"
;AT 18,18;"▌";AT 17,18;"▌";AT 16
,18;"▌"
 680 PRINT AT 21,21;"0▄▄▄▄▄▄▄▄5"
 690 PRINT AT C,B;"█"
 700 PRINT AT C,B;" "
 710 IF X>20 OR X<1 OR Y<1 OR Y>
30 THEN GOTO 810
 720 IF D=500 THEN GOTO 5
 730 PRINT AT X,Y;"O"
 740 PRINT AT X,Y;" "
 750 IF Y=(28) AND X=(10) THEN G
OTO 1500
 809 NEXT D
 810 CLS
 820 LET A=0
 830 FOR A=0 TO 31
 840 PRINT AT 9,A;"█"
 850 NEXT A
 860 PRINT
 870 PRINT
 880 PRINT AT 12,2;"VOCE SAIU FO
RA DO RADAR, BATEU EM RECIFES E
FOI A PIQUE"
 885 FOR M=0 TO 30
 890 NEXT M
 900 CLS
 910 GOTO 1
1500 CLS
1510 SLOW
1515 PRINT AT 0,13;"▄▄▄▄▄▄"
1530 PRINT AT 1,13;"▛    ▌"
1540 PRINT AT 2,2;"███████████████
████████████"
1550 FOR Z=2 TO 14
1560 PRINT AT Z,2;"█"
1570 PRINT AT Z,29;"█"
1580 NEXT Z
1585 PRINT AT 15,3;"█";AT 15,28;
"█";AT 16,4;"█";AT 16,27;"█";AT
17,4;"█";AT 17,27;"█";AT 18,4;"█
";AT 18,27;"█"
1590 PRINT AT 19,5;"██████████████
███████████"
1600 PRINT AT 20,9;"▌";AT 20,22;
"▌";AT 20,11;"PERISCOPIO"
1610 PRINT AT 16,1;" ▄▄ ";AT 16,2
8;" ▄▄ ";AT 17,1;"▐";AT 17,3;"▌";
AT 17,28;"▐";AT 17,30;"▌";AT 18,
1;"▐";AT 18,3;"▌";AT 18,28;"▐";A
T 18,30;"▌";AT 19,1;"▐";AT 19,3;
"▌";AT 19,28;"▐";AT 19,30;"▌";AT
 20,1;"▐";AT 20,3;"▌";AT 20,28;"
▐";AT 20,30;"▌";AT 21,1;"▐";AT 2
1,3;"▌";AT 21,28;"▐";AT 21,30;"▌
"
1620 LET A=2100
1630 LET L=0
1635 LET B=6
1640 LET X=24
1650 LET D=7
1660 LET O=24
1670 LET E=0
1680 LET H=7
```

→

```
1695 LET F=0
1700 LET G=0
1705 LET I=10
1710 LET Y=14
1715 LET J=0
1720 LET K=0
1725 PRINT AT 6,X;"      "
1730 LET O=11
1735 LET X=24
1745 LET A=A-100
1750 PRINT AT 3,15;"**"
1755 PRINT AT 1,14;A
1757 FOR R=0 TO 3
1758 NEXT R
1760 PRINT AT 1,14;"     "
1765 IF A<=0 THEN GOTO 3000
1775 PRINT AT 6,X;"▄▄█▄▄ "
1778 LET X=X-1
1780 LET O=O-1
1785 IF O=0 THEN GOTO 1725
1787 IF INKEY$="0" THEN GOTO 187
6
1790 LET D=D-(1 AND INKEY$="7")
1795 PRINT AT D,3;"//////////////
//////////////"
1797 PRINT AT 8,3;"//////////////
//////////////"
1800 PRINT AT 9,3;"//////////////
//////////////"
1805 PRINT AT 10,3;"/////////////
///////////////"
1810 PRINT AT 11,3;"/////////////
///////////////"
1815 PRINT AT 12,3;"/////////////
///////////////"
1820 PRINT AT 13,3;"/////////////
//////////////"
1825 PRINT AT 14,3;"/////////////
///////////////"
1830 PRINT AT 15,4;"/////////////
/////////////"
1835 PRINT AT 16,5;"/////////////
///////////"
1840 PRINT AT 17,5;"/////////////
///////////"
1845 PRINT AT 18,5;"/////////////
///////////"
1848 PRINT AT 18,H;"▛";AT 18,0;"
▜"
1850 IF D<=B THEN GOTO 2030
1853 PRINT AT 6,X;"▄▄█▄▄ "
1855 LET Y=Y+(3 AND INKEY$="8")-
(3 AND INKEY$="5")
1856 IF Y<=H OR Y>=0 THEN GOTO 1
745
1860 PRINT AT 18,Y;"/▄/"
1865 LET X=X-1
1875 GOTO 1745
1876 FOR P=17 TO 6 STEP -1
1890 PRINT AT P,Y+1;"."
1895 PRINT AT P,Y+1;" "
1900 NEXT P
1905 IF Y+1=X+3 THEN GOTO 1930
1910 LET E=E+1
1915 IF E=10 THEN GOTO 2005
1920 LET A=A-100
1925 GOTO 1745
1930 FOR C=0 TO 5
1935 PRINT AT 6,X;"▄▄█▄▄"
1940 PRINT AT 6,X;"      "
1945 NEXT C
1950 FOR F=0 TO 3
```

```
1955 PRINT AT 6-F,X;"▖ ";AT 6-F,
X+F;" █ ";AT 6-F,X-F;" ▜ ";AT 6-
F,X+2;" ▗";
1960 PRINT AT 6-F,X;"  ";AT 6-F,
X+F;"   ";AT 6-F,X-F;"   ";AT 6-
F,X+2;"  "
1965 NEXT F
1970 LET E=E+1
1975 LET G=G+1
1980 LET X=24
1985 LET O=11
1990 IF E=10 THEN GOTO 2005
1995 LET A=A-100
2000 GOTO 1745
2005 CLS
2010 PRINT AT 6,0;"TOTAL DE TIRO
S=";AT 6,16;E
2015 PRINT AT 10,0;"TOTAL DE BAR
COS DESTRUIDOS=";AT 10,28;G
2020 PAUSE 300
2025 GOTO 3095
2030 PRINT AT 6,X;"////"
2035 LET A=2100
2050 IF D=2 THEN GOTO 3080
2051 IF INKEY$="7" THEN GOTO 179
0
2053 LET D=D+(1 AND INKEY$="6")
2055 PRINT AT D,3;"
                 "
2057 IF D<=(6) THEN GOTO 2053
2060 GOTO 1790
2065 CLS
2070 FOR K=0 TO 31
2075 PRINT AT 9,K;"█"
2080 NEXT K
2085 PRINT AT 12,1;"A RESERVA DE
 AR EGOTOU-SE"
2090 PAUSE 300
2095 GOTO 3100
3000 CLS
3005 FOR M=0 TO 10
3010 PRINT AT 9,14;"████";AT 10,1
4;"████";AT 11,14;"████"
3015 PRINT AT 9,14;"    ";AT 10,1
4;"    ";AT 11,14;"    "
3020 NEXT M
3025 FOR L=0 TO 8
3030 PRINT AT 9-L,14;" ▛ ";AT 10
-L,14-L;"█";AT 9,14-L;"▜";AT 10+
L,14;"▗";AT 10,14+L;"▙"
3035 PRINT AT 9-L,14;"   ";AT 10
-L,14-L;"   ";AT 9,14-L;"   ";AT
10+L,14;"   ";AT 10,14+L;"   "
3040 NEXT L
3045 CLS
3050 FOR N=0 TO 31
3055 PRINT AT 9,N;"█"
3060 NEXT N
3065 PRINT AT 12,1;"SUBMARINO AL
VEJADO,FOI A PIQUE"
3067 FOR M=0 TO 30
3070 NEXT M
3075 GOTO 3095
3080 CLS
3082 FOR K=0 TO 31
3083 PRINT AT 9,K;"█"
3084 NEXT K
3085 PRINT AT 8,4;"O SUBMARINO S
UBMERGIU DEMAIS"
3087 FOR M=0 TO 30
3090 NEXT M
3095 CLS
3100 PRINT AT 10,14;"FIM"
```

Conexão

Desafie seu TK 85 para uma partida, e mostre ao seu equipamento que você pode superá-lo facilmente.

Características Gerais do Programa.

Após ter sido carregado o programa, perguntado o número de jogadores (1-2). Se a opção for "1", então você deve escolher se deseja iniciar o jogo ou não. Em seguida selecione qual o nível de dificuldade que deseja jogar.

O tabuleiro do jogo que será exibido em sua tela possui 6 linhas por 7 colunas, que estão representadas pelas letras de A até F.

Seu objetivo é completar uma sequência, não importando o tipo e sua direção, de no mínimo quatro pontos antes que seu competidor.

Se seu adversário optou pela coluna A por exemplo, quando for sua vez de jogar poderá optar pela mesma jogada de seu adversário com o objetivo de prejudicá-lo.

Cada jogador possui um placar, onde será registrado o número de vitórias que cada um conseguiu.

Se você estiver jogando contra o seu equipamento; e se for sua vez de jogar pressione a tecla "O"; o computador tomará a sua vez e jogará contra ele mesmo.

Notas sobre Programa

N$	Variável que acumula qualquer combinação da linha ganha.
Y$	Variável que determina as ordens de jogada
S1,S2	Placar de cada jogador
A-F	Representa os números 1 a 6, e auxilia o programa a rodar com mais rapidez.
P$	Variável que representa o número de jogadores.
S$	Determina quem irá iniciar a partida.
H$	Variável que determina o nível de dificuldade.
A$ até G$	Acumula as posições da área de jogo nas colunas de A até G.
I$	Seleciona as colunas de A até G ou 0.
1$	Os espaços da área de jogo.
J$	O string da coluna selecionada é copiada em J$, e o espaço acumulado em L$ é somado em J$ retornando astring da coluna selecionada.
Z	O primeiro número de J$ que não é um código de espacejamento.
R	Altera o valor para indicar uma linha ganha
Q$	Retém as linhas para comparação, para o computador realizar sua jogada.
Q	Realiza um loop, e o limite máximo de Q é determinado por H$.

As sete colunas que correpondem as letras de A até G, são acumuladas em A$ até G$, que consistem em seis espaços. O valor de Y$ e P$ é verificado para determinar quem será o próximo a jogar.

Depois que o jogador selecionar a coluna, uma verificação é feita para se saber se a tecla "0" foi selecionada. Em qualquer caso o programa salta para rotina de jogo.

Quando uma coluna é escolhida, o computador a verifica para certificar se esta ainda não foi preenchida antes de dar continuidade a jogada. Passada a informação para a coluna esta é copiada em J$, em seguida ajusta-se a mesma para que esta contenha o espaço adicional, e recopiada no string original.

Com este pedaço adicional, todas as 69 possibilidades de vitórias são reintegradas em N$. Durante a verificação se for encontrada uma linha que contenha uma seqüência de quatro, então R assume o valor de 1. Se não for encontrada nenhuma sequencia de quatro, então Y$ assume um novo valor, e o loop FOR NEXT de Q irá decidir qual o melhor movimento para o computador realizar sua próxima jogada. No caso, do seu adversário não ser o computador então este loop não será utilizado.

Listagem do programa Conexão

```
  10 DIM N$(69,4)
  20 LET Y$="1"
  30 LET S1=0
  35 LET I$=""
  40 LET S2=S1
  50 LET A=1
  60 LET B=A+A
  70 LET C=B+A
  80 LET D=C+A
  90 LET E=D+A
 100 LET F=E+A
 110 PRINT "NUMERO DE JOGADORES"
,,,"1 OU 2"
 120 LET P$=INKEY$
 130 IF P$="2" THEN GOTO 230
 140 IF P$<>"1" THEN GOTO 120
 150 PRINT ,,"VOCE DESEJA COMECA
R?(S/N)"
 160 LET S$=INKEY$
 170 IF S$="S" THEN GOTO 200
 180 IF S$<>"N" THEN GOTO 160
 190 LET Y$="2"
 200 PRINT ,,"SELECIONE O NIVEL
DE DIFICULDADE",,,"1, 2 OU 3"
 205 SLOW
 210 LET H$=INKEY$
 220 IF H$<CHR$ 29 OR H$>CHR$ 31
 THEN GOTO 210
 230 LET A$="      █"
 235 LET R=0
 240 LET B$=A$
 250 LET C$=A$
 260 LET D$=A$
 270 LET E$=A$
 280 LET F$=A$
 290 LET G$=A$
 300 CLS
```

```
 310 FOR J=C TO 16
 320 FOR K=8 TO 22 STEP B
 330 PRINT AT J,K;"▒"
 340 NEXT K
 350 NEXT J
 360 FOR J=D TO 16 STEP B
 370 FOR K=9 TO 21 STEP B
 380 PRINT AT J,K;"▒"
 390 NEXT K
 400 NEXT J
 410 PRINT TAB 9;"A B C D E F G"
 TAB 23;" OR 0" AND P$="1"
 420 PRINT AT 8,0;"JOGADOR1";TAB
 23;"MAQUINA" AND P$="1";"JOGADO
R2" AND P$="2";TAB 4;S1;TAB 28;S
2
 430 IF Y$="1" THEN GOTO 460
 440 IF P$="2" THEN GOTO 480
 450 IF Y$="2" THEN GOTO 1860
 460 PRINT AT 20,0;"JOGADOR 1 SU
A VEZ"
 470 GOTO 490
 480 PRINT AT 20,0;"JOGADOR 2 SU
A VEZ"
 485 SLOW

 490 IF A$(A)<>" " AND B$(A)<>"
" AND C$(A)<>" " AND D$(A)<>" "
AND E$(A)<>" " AND F$(A)<>" " AN
D G$(A)<>" " THEN GOTO 4700
 495 IF P$="1" AND (K=1 OR Y$="2
") THEN GOTO 560
```

```
 500 PRINT AT 19,9;"SELECIONE CO
LUNAS"
 510 LET I$=INKEY$
 515 SLOW
 520 IF I$="0" AND P$<>"2" THEN
GOTO 1860
 530 IF I$<CHR$ 38 OR I$>CHR$ 44
THEN GOTO 510
 540 GOSUB 560
 550 GOTO 510
 560 IF I$="A" AND A$(1)<>" " TH
EN RETURN
 570 IF I$="A" AND A$(1)<>" " TH
EN RETURN
 580 IF I$="B" AND B$(1)<>" " TH
EN RETURN
 590 IF I$="C" AND C$(1)<>" " TH
EN RETURN
 600 IF I$="D" AND D$(1)<>" " TH
EN RETURN
 610 IF I$="E" AND E$(1)<>" " TH
EN RETURN
 620 IF I$="F" AND F$(1)<>" " TH
EN RETURN
 630 IF I$="G" AND G$(1)<>" " TH
EN RETURN
 640 LET L$="O"
 650 IF Y$="1" THEN LET L$="*"
 660 IF I$="A" THEN LET J$=A$
 670 IF I$="B" THEN LET J$=B$
 680 IF I$="C" THEN LET J$=C$
 690 IF I$="D" THEN LET J$=D$ →
```

```
700 IF I$="E" THEN LET J$=E$
705 IF I$="F" THEN LET J$=F$
710 IF I$="G" THEN LET J$=G$
720 FOR Z=1 TO 7
730 IF J$(Z)<>" " THEN GOTO 750
740 NEXT Z
750 LET J$(Z-1)=L$
760 LET R=0
770 IF Y$="1" OR (Y$="2" AND P$
="2") THEN GOTO 1590
780 FAST
790 IF I$="A" THEN LET A$=J$
800 IF I$="B" THEN LET B$=J$
810 IF I$="C" THEN LET C$=J$
820 IF I$="D" THEN LET D$=J$
830 IF I$="E" THEN LET E$=J$
840 IF I$="F" THEN LET F$=J$
850 IF I$="G" THEN LET G$=J$
860 LET N$(1)=A$(A TO D)
870 LET N$(2)=B$(A TO D)
880 LET N$(3)=C$(A TO D)
890 LET N$(4)=D$(A TO D)
900 LET N$(5)=E$(A TO D)
910 LET N$(6)=F$(A TO D)
920 LET N$(7)=G$(A TO D)
930 LET N$(8)=A$(B TO E)
940 LET N$(9)=B$(B TO E)
950 LET N$(10)=C$(B TO E)
960 LET N$(11)=D$(B TO E)
970 LET N$(12)=E$(B TO E)
980 LET N$(13)=F$(B TO E)
990 LET N$(14)=G$(B TO E)
1000 LET N$(15)=A$(C TO F)
1010 LET N$(16)=B$(C TO F)
1020 LET N$(17)=C$(C TO F)
1030 LET N$(18)=D$(C TO F)
1040 LET N$(19)=E$(C TO F)
1050 LET N$(20)=F$(C TO F)
1060 LET N$(21)=G$(C TO F)
1070 LET N$(22)=A$(A)+B$(A)+C$(A
)+D$(A)
1080 LET N$(23)=A$(B)+B$(B)+C$(B
)+D$(B)
1090 LET N$(24)=A$(C)+B$(C)+C$(C
)+D$(C)
1100 LET N$(25)=A$(D)+B$(D)+C$(D
)+D$(D)
1110 LET N$(26)=A$(E)+B$(E)+C$(E
)+D$(E)
1120 LET N$(27)=A$(F)+B$(F)+C$(F
)+D$(F)
1130 LET N$(28)=B$(A)+C$(A)+D$(A
)+E$(A)
1140 LET N$(29)=B$(B)+C$(B)+D$(B
)+E$(B)
1150 LET N$(30)=B$(C)+C$(C)+D$(C
)+E$(C)
1160 LET N$(31)=B$(D)+C$(D)+D$(D
)+E$(D)
1170 LET N$(32)=B$(E)+C$(E)+D$(E
)+E$(E)
1180 LET N$(33)=B$(F)+C$(F)+D$(F
)+E$(F)
1190 LET N$(34)=C$(A)+D$(A)+E$(A
)+F$(A)
1200 LET N$(35)=C$(B)+D$(B)+E$(B
)+F$(B)
1210 LET N$(36)=C$(C)+D$(C)+E$(C
)+F$(C)
1220 LET N$(37)=C$(D)+D$(D)+E$(D
)+F$(D)
1230 LET N$(38)=C$(E)+D$(E)+E$(E
)+F$(E)
```

```
1240 LET N$(39)=C$(F)+D$(F)+E$(F
)+F$(F)
1250 LET N$(40)=D$(A)+F$(A)+F$(A
)+G$(A)
1260 LET N$(41)=D$(B)+E$(B)+F$(B
)+G$(B)
1270 LET N$(42)=D$(C)+E$(C)+F$(C
)+G$(C)
1280 LET N$(43)=D$(D)+E$(D)+F$(D
)+G$(D)
1290 LET N$(44)=D$(E)+E$(E)+F$(E
)+G$(E)
1300 LET N$(45)=D$(F)+E$(F)+F$(F
)+G$(F)
1310 LET N$(46)=D$(A)+E$(B)+F$(C
)+G$(D)
1320 LET N$(47)=D$(B)+E$(C)+F$(D
)+G$(E)
1330 LET N$(48)=D$(C)+E$(D)+F$(E
)+G$(F)
1340 LET N$(49)=C$(A)+D$(B)+E$(C
)+F$(D)
1350 LET N$(50)=C$(B)+D$(C)+E$(D
)+F$(E)
1360 LET N$(51)=C$(C)+D$(D)+E$(E
)+F$(F)
1370 LET N$(52)=B$(A)+C$(B)+D$(C
)+E$(D)
1380 LET N$(53)=B$(B)+C$(C)+D$(D
)+E$(E)
1390 LET N$(54)=B$(C)+C$(D)+D$(E
)+E$(F)
1400 LET N$(55)=A$(A)+B$(B)+C$(C
)+D$(D)
1410 LET N$(56)=A$(B)+B$(C)+C$(D
)+D$(E)
1420 LET N$(57)=A$(C)+B$(D)+C$(E
)+D$(F)
1430 LET N$(58)=A$(D)+B$(C)+C$(B
)+D$(A)
1440 LET N$(59)=A$(E)+B$(D)+C$(C
)+D$(B)
1450 LET N$(60)=A$(F)+B$(E)+C$(D
)+D$(C)
1460 LET N$(61)=B$(D)+C$(C)+D$(B
)+E$(A)
1470 LET N$(62)=B$(E)+C$(D)+D$(C
)+E$(B)
1480 LET N$(63)=B$(F)+C$(E)+D$(D
)+E$(C)
1490 LET N$(64)=C$(D)+D$(C)+E$(B
)+F$(A)
1500 LET N$(65)=C$(E)+D$(D)+E$(C
)+F$(B)
1510 LET N$(66)=C$(F)+D$(E)+E$(D
)+F$(C)
1520 LET N$(67)=D$(D)+E$(C)+F$(B
)+G$(A)
1530 LET N$(68)=D$(E)+E$(D)+F$(C
)+G$(B)
1540 LET N$(69)=D$(F)+E$(E)+F$(D
)+G$(C)
1550 FOR J=1 TO 69
1560 IF N$(J)=L$+L$+L$+L$ THEN L
ET R=1
1570 NEXT J
1580 IF Y$="1" OR P$="2" THEN GO
TO 1800
1590 SLOW
1595 FOR J=1 TO 5
1596 NEXT J
1600 FOR I=1 TO Z-1
1610 PRINT AT I*2+3,7+(CODE I$-3
7)*2;L$
```

```
1620 NEXT I
1622 LET K=0
1624 FOR J=1 TO 5
1625 NEXT J
1630 IF Y$="1" THEN GOTO 780
1635 IF P$="2" THEN GOTO 780
1640 GOTO 1800
1650 SLOW
1655 IF Y$="1" THEN GOTO 1730
1660 PRINT AT 20,0;"VENCEU O COM
PUTADOR      "
1670 IF P$="2" THEN GOTO 1690
1680 GOTO 1700
1690 PRINT AT 20,0;"VENCEU O JOG
ADOR 2      "
1700 LET S2=S2+1
1710 LET Y$="1"
1720 GOTO 1760
1730 PRINT AT 20,0;"VENCEU O JOG
ADOR 1      "
1740 LET S1=S1+1
1750 LET Y$="2"
1760 FOR J=1 TO 100
1770 NEXT J
1775 CLS
1780 IF P$="2" THEN GOTO 230
1790 GOTO 200
1800 IF R=1 THEN GOTO 1650
1810 IF Y$="1" THEN GOTO 1840
1820 LET Y$="1"
1830 GOTO 430
1840 LET Y$="2"
1850 GOTO 430
1860 DIM O$(4,4)
1865 IF I$="0" THEN LET K=1
1870 FAST
1880 FOR Q=1 TO VAL H$*2
1890 IF I$="0" THEN GOTO 1930
1900 IF Q=1 OR Q=4 OR Q=6 THEN L
ET L$="O"
1910 IF Q=2 OR Q=3 OR Q=5 THEN L
ET L$="*"
1920 GOTO 1950
1930 IF Q=1 OR Q=4 OR Q=6 THEN L
ET L$="*"
1940 IF Q=2 OR Q=3 OR Q=5 THEN L
ET L$="O"
1950 IF Q>2 THEN GOTO 2050
2010 LET O$(A)=" "+L$+L$+L$
2020 LET O$(B)=L$+" "+L$+L$
2030 LET O$(C)=L$+L$+" "+L$
2040 LET O$(D)=L$+L$+L$+" "
2050 IF Q<3 THEN GOTO 2140
2060 LET O$(A)=" "+L$+" "+L$
2070 LET O$(B)=L$+" "+L$+" "
2080 LET O$(C)=L$+L$+" "+" "
2085 LET O$(D)=" "+L$+L$+" "
2090 IF Q<5 THEN GOTO 2140
2100 LET O$(A)=" "+L$+" "+" "
2110 LET O$(B)=" "+" "+L$+" "
2120 LET O$(C)=" "+" "+" "+L$
2130 LET O$(D)=" "+" "+L$+" "
2140 IF N$(A)=O$(A) OR N$(8)=O$(
A) OR N$(15)=O$(A) THEN GOTO 400
0
2150 IF A$(B)<>" " AND (N$(22)=O
$(A) OR N$(55)=O$(A)) THEN GOTO
4000
2160 IF A$(C)<>" " AND (N$(23)=O
$(A) OR N$(56)=O$(A)) THEN GOTO
4000
2170 IF A$(D)<>" " AND (N$(24)=O
$(A) OR N$(57)=O$(A)) THEN GOTO
4000
2180 IF A$(E)<>" " AND (N$(25)=O
$(A) OR N$(58)=O$(A)) THEN GOTO
4000
2190 IF A$(F)<>" " AND (N$(26)=O
$(A) OR N$(59)=O$(A)) THEN GOTO
4000
2200 IF N$(27)=O$(A) OR N$(60)=O
$(A) THEN GOTO 4000
2210 IF N$(B)=O$(A) OR N$(9)=O$(
A) OR N$(16)=O$(A) THEN GOTO 410
0
2220 IF B$(B)<>" " AND (N$(28)=O
$(A) OR N$(52)=O$(A)) THEN GOTO
4100
2230 IF B$(C)<>" " AND (N$(29)=O
$(A) OR N$(53)=O$(A)) THEN GOTO
4100
2240 IF B$(D)<>" " AND (N$(30)=O
$(A) OR N$(54)=O$(A)) THEN GOTO
4100
2250 IF B$(E)<>" " AND (N$(31)=O
$(A) OR N$(61)=O$(A)) THEN GOTO
4100
2260 IF B$(F)<>" " AND (N$(32)=O
$(A) OR N$(62)=O$(A)) THEN GOTO
4100
2270 IF N$(33)=O$(A) OR N$(63)=O
$(A) THEN GOTO 4100
2280 IF N$(C)=O$(A) OR N$(10)=O$
(A) OR N$(17)=O$(A) THEN GOTO 42
00
2290 IF C$(B)<>" " AND (N$(34)=O
$(A) OR N$(49)=O$(A)) THEN GOTO
4200
2300 IF C$(C)<>" " AND (N$(35)=O
$(A) OR N$(50)=O$(A)) THEN GOTO
4200
2310 IF C$(D)<>" " AND (N$(36)=O
$(A) OR N$(51)=O$(A)) THEN GOTO
4200
2320 IF C$(E)<>" " AND (N$(37)=O
$(A) OR N$(64)=O$(A)) THEN GOTO
4200
2330 IF C$(F)<>" " AND (N$(38)=O
$(A) OR N$(65)=O$(A)) THEN GOTO
4200
2340 IF N$(39)=O$(A) OR N$(66)=O
$(A) THEN GOTO 4200
2350 IF N$(D)=O$(A) OR N$(11)=O$
(A) OR N$(18)=O$(A) THEN GOTO 43
00
2360 IF D$(B)<>" " AND (N$(40)=O
$(A) OR N$(46)=O$(A)) THEN GOTO
4300
2370 IF D$(C)<>" " AND (N$(41)=O
$(A) OR N$(47)=O$(A)) THEN GOTO
4300
2380 IF D$(D)<>" " AND (N$(42)=O
$(A) OR N$(48)=O$(A)) THEN GOTO
4300
2390 IF D$(E)<>" " AND (N$(43)=O
$(A) OR N$(67)=O$(A)) THEN GOTO
4300
2400 IF D$(F)<>" " AND (N$(44)=O
$(A) OR N$(68)=O$(A)) THEN GOTO
4300
2410 IF N$(45)=O$(A) OR N$(69)=O
$(A) THEN GOTO 4300
2420 IF N$(E)=O$(A) OR N$(12)=O$
(A) OR N$(19)=O$(A) THEN GOTO 44
00
2430 IF N$(F)=O$(A) OR N$(13)=O$
(A) OR N$(20)=O$(A) THEN GOTO 45
00
```

```
2440 IF N$(7)=0$(A) OR N$(14)=0$
(A) OR N$(21)=0$(A) THEN GOTO 46
00
2450 IF B$(B)<>" " AND (N$(22)=0
$(B) OR N$(55)=0$(B)) THEN GOTO
4100
2460 IF B$(C)<>" " AND (N$(23)=0
$(B) OR N$(55)=0$(B)) THEN GOTO
4100
2470 IF B$(D)<>" " AND (N$(24)=0
$(B) OR N$(56)=0$(B)) THEN GOTO
4100
2480 IF B$(E)<>" " AND (N$(25)=0
$(B) OR N$(57)=0$(B)) THEN GOTO
4100
2490 IF B$(F)<>" " AND (N$(26)=0
$(B) OR N$(60)=0$(B)) THEN GOTO
4100
2500 IF N$(27)=0$(B) THEN GOTO 4
100
2510 IF C$(B)<>" " AND N$(28)=0$
(B) THEN GOTO 4200
2520 IF C$(C)<>" " AND (N$(29)=0
$(B) OR N$(52)=0$(B)) THEN GOTO
4200
2530 IF C$(D)<>" " AND (N$(30)=0
$(B) OR N$(61)=0$(B)) THEN GOTO
4200
2540 IF C$(E)<>" " AND (N$(31)=0
$(B) OR N$(62)=0$(B)) THEN GOTO
4200
2550 IF C$(F)<>" " AND (N$(32)=0
$(B) OR N$(63)=0$(B)) THEN GOTO
4200
2560 IF N$(33)=0$(B) THEN GOTO 4
200
2570 IF D$(B)<>" " AND N$(34)=0$
(B) THEN GOTO 4300
2580 IF D$(C)<>" " AND (N$(35)=0
$(B) OR N$(49)=0$(B)) THEN GOTO
4300
2590 IF D$(D)<>" " AND (N$(36)=0
$(B) OR N$(50)=0$(B) OR N$(64)=0
$(B)) THEN GOTO 4300
2600 IF D$(E)<>" " AND (N$(37)=0
$(B) OR N$(51)=0$(B) OR N$(65)=0
$(B)) THEN GOTO 4300
2610 IF D$(F)<>" " AND (N$(38)=0
$(B) OR N$(66)=0$(B)) THEN GOTO
4300
2620 IF N$(39)=0$(B) THEN GOTO 4
300
2630 IF E$(B)<>" " AND N$(40)=0$
(B) THEN GOTO 4400
2640 IF E$(C)<>" " AND (N$(41)=0
$(B) OR N$(46)=0$(B)) THEN GOTO
4400
2650 IF E$(D)<>" " AND (N$(42)=0
$(B) OR N$(47)=0$(B) OR N$(67)=0
$(B)) THEN GOSUB 4400
2660 IF E$(E)<>" " AND (N$(43)=0
$(B) OR N$(48)=0$(B) OR N$(68)=0
$(B)) THEN GOTO 4400
2670 IF E$(F)<>" " AND (N$(44)=0
$(B) OR N$(69)=0$(B)) THEN GOTO
4400
2680 IF N$(45)=0$(B) THEN GOTO 4
400
2690 IF C$(B)<>" " AND N$(22)=0$
(C) THEN GOTO 4200
2700 IF C$(C)<>" " AND (N$(23)=0
$(C) OR N$(58)=0$(C)) THEN GOTO
4200
2710 IF C$(D)<>" " AND (N$(24)=0
$(C) OR N$(59)=0$(C)) THEN GOTO
4200
2720 IF C$(E)<>" " AND (N$(25)=0
$(C) OR N$(56)=0$(C) OR N$(60)=0
$(C)) THEN GOTO 4200
2730 IF C$(F)<>" " AND (N$(26)=0
$(C) OR N$(57)=0$(C)) THEN GOTO
4200
2740 IF N$(27)=0$(C) THEN GOTO 4
200
2750 IF D$(B)<>" " AND N$(28)=0$
(C) THEN GOTO 4300
2760 IF D$(C)<>" " AND (N$(29)=0
$(C) OR N$(61)=0$(C)) THEN GOTO
4300
2770 IF D$(D)<>" " AND (N$(30)=0
$(C) OR N$(52)=0$(C) OR N$(62)=0
$(C)) THEN GOTO 4300
2780 IF D$(E)<>" " AND (N$(31)=0
$(C) OR N$(53)=0$(C) OR N$(63)=0
$(C)) THEN GOTO 4300
2790 IF D$(F)<>" " AND (N$(32)=0
$(C) OR N$(54)=0$(C)) THEN GOTO
4300
2800 IF N$(33)=0$(C) THEN GOTO 4
300
2810 IF E$(B)<>" " AND N$(34)=0$
(C) THEN GOTO 4400
2820 IF E$(C)<>" " AND (N$(35)=0
$(C) OR N$(64)=0$(C)) THEN GOTO
4400
2830 IF E$(D)<>" " AND (N$(36)=0
$(C) OR N$(49)=0$(C) OR N$(65)=0
$(C)) THEN GOTO 4400
2840 IF E$(E)<>" " AND (N$(37)=0
$(C) OR N$(50)=0$(C) OR N$(66)=0
$(C)) THEN GOTO 4400
2850 IF E$(F)<>" " AND (N$(38)=0
$(C) OR N$(51)=0$(C)) THEN GOTO
4400
2860 IF N$(39)=0$(C) THEN GOTO 4
400
2870 IF F$(B)<>" " AND N$(40)=0$
(C) THEN GOTO 4500
2880 IF F$(C)<>" " AND (N$(41)=0
$(C) OR N$(67)=0$(C)) THEN GOTO
4500
2890 IF F$(D)<>" " AND (N$(42)=0
$(C) OR N$(46)=0$(C) OR N$(68)=0
$(C)) THEN GOTO 4500
2900 IF F$(E)<>" " AND (N$(43)=0
$(C) OR N$(47)=0$(C) OR N$(69)=0
$(C)) THEN GOTO 4500
2910 IF F$(F)<>" " AND (N$(44)=0
$(C) OR N$(48)=0$(C)) THEN GOTO
4500
2920 IF N$(45)=0$(C) THEN GOTO 4
500
2930 IF D$(B)<>" " AND (N$(22)=0
$(D) OR N$(58)=0$(D)) THEN GOTO
4300
2940 IF D$(C)<>" " AND (N$(23)=0
$(D) OR N$(59)=0$(D)) THEN GOTO
4300
2950 IF D$(D)<>" " AND (N$(24)=0
$(D) OR N$(60)=0$(D)) THEN GOTO
4300
2960 IF D$(E)<>" " AND (N$(25)=0
$(D) OR N$(55)=0$(D)) THEN GOTO
4300
2970 IF D$(F)<>" " AND (N$(26)=0
$(D) OR N$(56)=0$(D)) THEN GOTO
4300
2980 IF N$(27)=0$(D) OR N$(57)=0
$(D) THEN GOTO 4300
2990 IF E$(B)<>" " AND (N$(28)=0
```

```
$(D) OR N$(61)=0$(D)) THEN GOTO
4300
3000 IF E$(C)<>" " AND (N$(29)=0
$(D) OR N$(62)=0$(D)) THEN GOTO
4400
3010 IF E$(D)<>" " AND (N$(30)=0
$(D) OR N$(63)=0$(D)) THEN GOTO
4400
3020 IF E$(E)<>" " AND (N$(31)=0
$(D) OR N$(52)=0$(D)) THEN GOTO
4400
3030 IF E$(F)<>" " AND (N$(32)=0
$(D) OR N$(53)=0$(D)) THEN GOTO
4400
3040 IF N$(33)=0$(D) OR N$(54)=0
$(D) THEN GOTO 4400
3050 IF F$(B)<>" " AND (N$(34)=0
$(D) OR N$(64)=0$(D)) THEN GOTO
4500
3060 IF F$(C)<>" " AND (N$(35)=0
$(D) OR N$(65)=0$(D)) THEN GOTO
4500
3070 IF F$(D)<>" " AND (N$(36)=0
$(D) OR N$(66)=0$(D)) THEN GOTO
4500
3080 IF F$(E)<>" " AND (N$(37)=0
$(D) OR N$(49)=0$(D)) THEN GOTO
4500
3090 IF F$(F)<>" " AND (N$(38)=0
$(D) OR N$(50)=0$(D)) THEN GOTO
4500
3100 IF N$(39)=0$(D) OR N$(51)=0
$(D) THEN GOTO 4500
3890 IF G$(B)<>" " AND (N$(40)=0
$(D) OR N$(67)=0$(D)) THEN GOTO
4600
3900 IF G$(C)<>" " AND (N$(41)=0
$(D) OR N$(68)=0$(D)) THEN GOTO
4600
3910 IF G$(D)<>" " AND (N$(42)=0
$(D) OR N$(69)=0$(D)) THEN GOTO
4600
3920 IF G$(E)<>" " AND (N$(43)=0
$(D) OR N$(46)=0$(D)) THEN GOTO
4600
3930 IF G$(F)<>" " AND (N$(44)=0
$(D) OR N$(47)=0$(D)) THEN GOTO
4600
3940 IF N$(45)=0$(D) OR N$(48)=0
$(D) THEN GOTO 4600
3950 NEXT Q
3960 LET T=INT (RND*7)
3970 GOTO 4000+100*T
4000 LET I$="A"
4010 GOSUB 490
4100 LET I$="B"
4110 GOSUB 490
4200 LET I$="C"
4210 GOSUB 490
4300 LET I$="D"
4310 GOSUB 490
4400 LET I$="E"
4410 GOSUB 490
4500 LET I$="F"
4510 GOSUB 490
4600 LET I$="G"
4610 GOSUB 490
4620 GOTO 3960
4700 PRINT AT 20,0;"A DRAW
"
4705 SLOW
4710 GOTO 1760
9000 CLEAR
9010 SAVE "CONECCAO "
9020 GOTO 1
```

Transferindo truques e programas em BASIC e Assembly do TK 85 para o TK 90X e vice-versa

Álvaro A. L. Domingues

Desde o seu lançamento, a Microhobby sempre mostrou vários programas e truques de programação para o TK 85 e seus compatíveis. Agora, ao lado do TK 85, estão aparecendo alguns programas para o TK 90X. Mas será que os usuários de ambos equipamentos, podem aproveitar todas as dicas e programas que vêm sendo publicados?

Os conceitos fundamentais de programação que aparecem em qualquer artigo são de extrema importância para usuários de todas as linhas de computadores. O que torna um artigo ou programas específicos para um equipamento são as suas particularidades. Existem diferentes filosofias de projeto, o que diversifica as soluções de hardware e software. Mas quantas vezes não deixamos de lado um programa interessante para um Apple só porque possuímos um TK 85? Imagine você se descobrisse, depois de recusar um programa desses, que a única diferença entre os equipamentos era apenas uma linha que continha um HOME e que poderia facilmente ser substituída por um CLS?...

Semelhanças e diferenças

Se você já teve oportunidade de operar um TK 90X, certamente percebeu que ele é bastante superior ao 85. Possui maior velocidade de processamento, cores, alta-resolução, caracteres gráficos especiais definidos pelo usuário e um número muito maior de funções, instruções e comandos.

Entretanto, existem algumas semelhanças que merecem ser apontadas. Em primeiro lugar, o microcomputador é o mesmo, um Z 80A. Segundo, a linguagem empregada é a mesma, o BASIC. O TK 90X possui quase todos os comandos, funções e instruções do TK 85, acrescidos dos recursos que o caracterizam. As únicas instruções que não estão disponí veis são o SCROLL e o UNPLOT.

Tabela I - Conversões do BASIC do TK 85 para o TK 90X

TK 85	TK 90X	Comentários
SCROLL	Automático	Se o programa foi desenvolvido visando o uso de efeitos especiais de SCROLL, substitua por RAND USR 3582 ou LET X=USR 3582.
PLOT X,Y	PRINT AT 21-Y/2, X/2	Coloque, após o ponto e vírgula, o caractere gráfico especial apropriado (teclas 1,2, 4 ou 7).
UNPLOT X,Y	PRINT AT 21-Y/2, X/2;" "	

O PLOT do TK 90X é diferente do TK 85, uma vez que está preparado para alta-resolução. Portanto, não pode ser substituído diretamente.

Do 90 para o 85

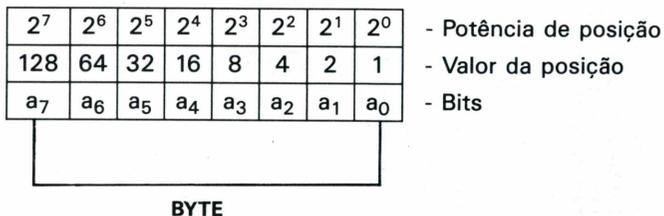
Alguns comandos, funções e instruções do TK 90X não estão disponíveis no TK 85 (Tabela II), mas podem ser traduzidos sem muitos problemas (evidentemente com as limitações inerentes ao TK 85). Dois deles merecem destaque especial: BIN e READ/DATA/RESTORE.

O primeiro, permite que um número seja transformado em decimal. A sua função é facilitar a colocação de instruções em Linguagem de Máquina ou a criação de caracteres especiais (não permitido no 85).

O número binário deve ter 8 bits (tamanho da palavra nos dois computadores). Na figura 1, podemos ver como se efetua esta transformação manualmente.

A tradução do READ/DATA/RESTORE é mais complicada e requer o uso de uma sub-rotina.

Figura 1: Estrutura de um byte.

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	- Potência de posição
128	64	32	16	8	4	2	1	- Valor da posição
a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	- Bits

BYTE

a_7 ... a_0 - bits (assumem valores zero ou 1)

Valor decimal do byte $= a_7 \times 2^7 + a_6 \times 2^6 \ldots + a_0 \times 2^0$
ou $= a_7 \times 128 + a_6 \times 64 \ldots + a_0$

As instruções READ/DATA permitem que uma série de dados sejam armazenados no próprio programa (READ lê estes dados, e DATA os armazena). RESTORE permite que um mesmo conjunto de dados seja usado mais de uma vez.

→

HOBBYSHOP VEJA SE SUA CIDADE TEM O QUE VOCÊ PRECISA

SÃO PAULO

MICRO service

Inclusão de 24 novas funções (Read, Data, etc.), Slow, High Speed, Alta Resolução, Porta de I/0, etc. para micro de tecnologia SINCLAIR ZX81.
Manutenção de microcomputadores SINCLAIR (TK 82, 83, 85, etc.) e TRS.
Wilson de Assis — Tel.: 203-7967

TKSOM-TKMORSE

2 Software de alta qualidade para Micros Sinclair com 16 K
TKSOM — coloca som no seu micro; contém 6 músicas; você pode programar suas músicas.
TKMORSE — lista sua mensagem em código morse; transmite sinais sonoros de mensagem pré-gravada; transmite sinais sonoros simultaneamente com a digitação.
Preço até 30-06-85 Cr$ 28.000
Envie cheque nominal para:
MARCIO ACCIOLY
Rua Dr. Saboia de Medeiros, 199-54 — Cep 04120 — São Paulo — SP
e receba os 2 Software pelo correio, sem mais nenhuma despesa.
PREÇOS ESPECIAIS PARA REVENDEDORES.

Transforme seu TK 85.
O mesmo efeito dos monitores de vídeo.
Fundo: preto
Letras: brancas
Com uma simples modificação no microcomputador.

TRANSVIDEO Fone: (011) 522-8100

Comércio de Computadores Ltda.
TK85 x TK2000?
Só na ENG você adquiri o seu TK2000 nas melhores condições e ainda dá o seu velho TK83, TK85 ou CP200 como parte de pagamento. TK2000 é na ENG. Showroom — Tel. 813-7570. Av. dos Tajurás, 406 — CEP: 05670.

apple cursos

CURSOS DIRIGIDOS DE MICRO-COMPUTADORES

- BASIC I e II e Applesoft
- ASSEMBLER 6502
- EDITOR DE TEXTO E PLANILHA ELETRÔNICA

NOVAS TURMAS (c/ 12 alunos)
INÍCIO IMEDIATO

Reservas pelos Telefones: 853-9457 — 853-2408 Rua Suzano, 78 — Jardim Paulista — São Paulo

O MELHOR SERVIÇO PELO MENOR PREÇO!

Faça como os funcionários da SABESP, BURI, KIBON e outros. Venha desvendar o computador da DATA RECORD INFORMÁTICA.
COBOL — BASIC — DIGITAÇÃO
Turmas especiais para crianças de 8 a 14 anos. (BOLSAS DE ESTUDO)
Av. Santo Amaro, 5.450 — Tel. 543-9937 — Brooklin — (em frente ao E.C. Banespa).

QUAL A INTERFACE QUE ESTÁ FALTANDO NO SEU MICRO?

É AQUELA QUE LHE DEVOLVERÁ O PRAZER DE FICAR EM FRENTE DO SEU MONITOR POR TEMPO ILIMITADO.

MICROTELA possibilita que você continue com seu TV, pois possue a mesma tela de poliester utilizada nos monitores de última geração, filtrando e eliminando os reflexos, ao mesmo tempo que aumenta a resolução da imagem.
Adicionalmente proporciona o mesmo efeito repousante dos monitores de fósforo colorido, utilizando acrílico nas tonalidades verde e ambar.

Informações com **MASTER STING LTDA.**
Caixa Postal 18708 — São Paulo — SP

SUPRIMENTOS P/ INFORMÁTICA

*FORMULÁRIOS
*DISKETES *FITAS IMPRESSORAS
*PAPEL XEROGRÁFICO
*SUPRIMENTOS P/ TELEX E ESCRITÓRIO
INFORMAX-PRODUTOS P/ INFORMÁTICA LTDA.
R. Domingos de Morais, 254-6º and. Cj. 602-A
Tel. (011) **570.7570**

SÃO JOSÉ DOS CAMPOS

EKTRONIC — COMPONENTES E SISTEMAS LTDA.

"SOFT-LOADER" — Interface micro-cassete para TK 82-C, 83, 85 e Ringo. Indica nível certo para carregar programas sem problemas e falhas. (Veja Microhobby Nº 10, 12 ou 13). Já um GRANDE SUCESSO PROVADO por centenas de usuários do TK. PREÇO: Cr$ 49.000,00 (Março).
Mande seu pedido com cheque nominal ou vale postal para EKTRONIC COMPONENTES E SISTEMAS LTDA. Caixa Postal 7004. São José dos Campos. CEP: 12200. Tel.: (0123) 291148.

BAHIA

Sua empresa poderia estar aqui.
Anuncie no HOBBYSHOP e todos os Leitores da região conhecerão sua empresa. Anúncio econômico e de retorno garantido.

RIO DE JANEIRO

PROSERV-Processamento Dados.Cursos e Rep.Ltda.
.MICROCOMPUTADORES (Novos e Usados)
.CURSOS (Cobol. Basic. CP/M. DBase II)
.SUPRIMENTOS (Formularios. Disquetes. Fitas. etc.)
.LIVROS E REVISTAS
.SOFTWARE (TRS80. Apple. TK85)
Lg.Nove de Abril 27 salas 626/628
Tel: (0243) 429800 - V.Redonda - RJ

MINAS GERAIS

MICRO E VIDEO

Curso de Basic com turmas mensais
Programas para toda linha de microcomputadores — Sinclair, TRS-80, Apple, TRS Color, Comodore CP/M — Aplicativos e Jogos (Solicite catálogo especificando seu equipamento).
Livros e revistas nacionais e estrangeiros. Venda de Micros, periféricos e suprimentos. Soft House.
VILLABELLA SHOPPING — LOJA 6
Avenida Japão, 229 — Cariru — CEP 35160 — Fone (031) 821-2888 — Ipatinga — MG.

Tabela V - Instruções que não podem ser traduzidas	
Instrução	Comentário
SOUND	Gera um sinal sonoro no alto-falante da TV
BORDER	Escolhe a cor da margem da tela
PAPER	Escolhe a cor da tela ("papel")
INK	Escolhe a cor da "tinta"
OVER	Permite a sobre-exibição no vídeo, de dois caracteres, sem que nenhum deles seja apagado
UDG	Permite a criação de caracteres gráficos especiais
ATTR	Fornece um valor para cada situação de um caractere na tela
VAL$	Assume como string o conteúdo de uma variável representada como um conjunto de caracteres em outra string
CIRCLE	Desenha um círculo na tela
IN/OUT	Controla a entrada (IN) e saída (OUT) de dados para periféricos adicionais
TRACE	Permite ao usuário acompanhar o desenrolar do programa, mostrando o número da linha que está sendo executada.
MERGE	Permite a união de dois programas BASIC na memória
POINT	Indica a situação de um ponto na tela
DRAW	Desenha uma reta de acordo com coordenadas específicas
PLOT	Funciona de maneira diferente do TK 85 (alta resolução)
LINE	Especifica a linha a ser executada num programa com auto-RUN.
OPEN/CLOSE/MOVE/ERASE/ CAT/FORMAT	Instruções reservadas para periféricos

Outras características

Cabe ainda chamar atenção para algumas características especiais do TK 90X. Uma delas é a presença de dois pontos, separando duas instruções, na mesma linha. Isso não é permitido no TK 85, o que nos leva a usar mais de uma linha na tradução de linhas com dois pontos. Este é um cuidado especial que deve ser tomado com as instruções IF/THEN.

O TK 90X possui também alguns caracteres diferentes, tais como colchetes,chaves e letras minúsculas, que não estão disponíveis no TK 85. Isso pode ser contornado, simplesmente substituindo estes caracteres ou eliminando-os do programa.

Se você pretende traduzir um programa em Linguagem de Máquina do 85 para o 90X, pode ter algumas surpresas. Uma delas é a impossibilidade de rodar um jogo porque ele ficou rápido demais. Isso ocorre porque o tempo de processamento do TK 90X (por uma série de razões) é muito mais rápido que o TK 85. Se você conhece bem o Assembly, poderá implementar loops "vazios" (que não executam nada) para retardar um pouco o andamento do programa.

Tabela II - Instruções e funções do TK 90X traduzíveis para o TK 85 .

TK 90x	**Exemplo**	**TK 85**	**Comentários**
BIN	LET A=BIN 10010101	LET A= nro. decimal. O nro. deve ser equivalente à conversão de binário em decimal.	A função BIN permite converter um número binário em decimal. Esta conversão é feita da seguinte forma: multiplica-se o valor de cada bit (0 ou 1) pelo valor de sua posição no byte, somando-se em seguida, todos os valores (ver figura 1).
READ/DATA/ RESTORE	READ X,Y DATA 90,10 RESTORE	Ver texto	READ/DATA é um par de instruções usado para armazenar dados no interior do programa. RESTORE restaura estes dados para nova leitura. (Ver texto).
DEF FN/FN	DEF FN A(X) = X*2+3 LET T = FN(2)	LET X$= "X*2+3" LET X=2 LET T=VAL X$	DEF FN permite definir uma função (FN).
SCREEN$	LET A$ SCREEN$	LET A$=PEEK (PEEK 16384+256* PEEK 16 387+1+Y+33*X).	Pesquisa caracteres na memória de vídeo.
	LET A=2 3	LET A=2**3	Operação de potenciação.
FLASH	PRINT AT 0,0; FLASH 1; "MENSAGEM"; FLASH O	FOR I=1 TO 10 PRINT AT 0, 0; "MENSAGEM" PRINT AT 0,0; "MENSAGEM" (caracteres invertidos) NEXT I	Faz a mensagem piscar na tela.
INVERSE	PRINT INVERSE 1; "MENSAGEM"	Passar o cursor para G e digitar a mensagem com os caracteres inversos	Exibe as letras invertidas na tela

Usando a linha REM

Quem está acostumado a digitar programas em Linguagem de Máquina no TK 85 já deve conhecer o uso da linha REM como forma de armazenamento de rotina em Assembly. O que o usuário não sabe é que está criando na realidade uma linha de DATA. Podemos então aproveitar estes conhecimentos de Linguagem de Máquina para a nossa tradução. O que devemos fazer é colocar os dados separados por vírgula como na linha DATA normal e fazer com que estes dados sejam lidos por PEEKS. Quem co-

nhece bem a estrutura da linha 1 REM sabe que no endereço 16513 está a palavra chave, REM, cujo código é 234. O código da vírgula (,) é 26 e deverá ser checado para que se possa verificar o final da operação.

Precisamos "envelopar" os dados de forma a permitir a sua fácil leitura e identificação pelo programa. Escolhemos o caractere gráfico (cursor G) SHIFT+A, que possui o código 8. Ele deverá aparecer antes do primeiro dado e após o último.

Para sabermos se o último dado foi lido, devemos checar o código de final de linha que, no TK 85, é 118 (NEW LINE).

Da forma apresentada, são permitidos dados númericos inteiros e positivos, mas o programa pode ser adaptado para leitura de dados decimais, negativos ou de STRINGs. Na listagem 1 são mostradas a sub-rotina na READ e uma rotina de teste (linhas 20 a 30). Repare na linha 10. Ela inicializa o contador de dados. Toda vez que precisarmos de uma nova leitura dos mesmos dados, teremos que acrescentá-la ao programa. Ela tem a mesma função do RESTORE, só que deve ser adicionada sempre antes da primeira leitura, já que tem a função de inicializar o contador de dados.

Truques do TK 85 para o TK 90X

Os leitores da Microhobby e usuários do TK 85 acostumaram-se a usar uma série de truques fornecidos pela revista. Quem não sabe "zerar" a primeira linha do programa ou reservar uma área na RAMTOP? Isto também é possível no TK 90X, conforme você pode observar na tabela III.

Tabela III - Alguns truques do TK 85 transferidos para o TK 90X

Função	TK 85	TK 90X
Zera a primeira	POKE 16510,0	POKE 23754,0
linha do programa		
Zera variável FRAMES	POKE 16436,255 POKE 16437,25	POKE 23672,0 POKE 23673,0
Reserva uma área na RAM-TOP) TOP) de X bytes.	POKE 16388,X-256 * INT (X/256) POKE 16389,INT(X/256)	CLEARX

Zerar a primeira linha do programa é útil para evitar que esta seja editada por acidente, fazendo-se perder, por exemplo, uma listagem em Linguagem de Máquina no TK 85, ou uma linha DATA com dados que não podem ser alterados no TK 90X.

A proteção de um programa na RAMTOP permite preservar um utilitário de uso geral de um NEW, uma vez que desejamos usá-lo em vários programas. Isso ocorre, por exemplo, num remunerador de linhas. Podemos renume rar um programa corrente, mas também querer carregá-lo a partir de uma fita. A operação de LOAD costuma destruir o programa que está na memória, impedindo este tipo de operação.

A variável FRAMES conta quadros de imagem da TV a partir do momento que o computador é ligado. Esta variável pode ser usada para contar o tempo disponível num jogo ou para gerar números aleatórios, entre outras coisas.

Por falar em Linguagem de Máquina...

Resta-nos fornecer os endereços do sistema. Na tabela IV apresentamos os endereços correspondentes em ambas as máquinas. A descrição do que faz cada uma delas está no manual do equipamento (capítulo 28, para o TK 85 e apêndice C, para o TK 90X). O uso destas variáveis é bastante conhecido por quem trabalha com Linguagem de Máquina. Explicar cada uma delas escapa as pretensões deste artigo. Uma fonte de consulta muito boa é o livro "Linguagem de Máquina para o TK", publicado pelas editoras Moderna/Micromega, destinado aos usuários do TK 85.

Tabela IV - Variáveis do sistema

Variável	TK 85	TK 90X
BREG	16414	23655
CDFLAG	16443	sem equivalência
CH ADD	16406	23645
COORDS	16438	23677
COORDS (byte 2)	16439	23678
DEST	16402	23629
DEF CC	16398	23684
D FILE	16396	sem equivalência
DF SZ	16418	23659
E LINE	16404	23641
ERR NR	16384	23610
E PPC	16294	23625
ERR SP	16386	23613
FLAGS	16385	23611
FLAGX	16429	23655
FRAMES	16436	23672
LAST K	16421	23560
MARGIN	16424	sem equivalência
MEM	16415	23656
MEMBOTT	16477	23698
MODE	16390	23617
NXTLIN	16425	23677
OLDPCC	16427	23662
PPC	16391	23621
PRBUFF	16444	23296
PR CC	16440	23680
RAMTOP	16388	23730
SEED	16434	23670
S POSN	16441	23688
S POS (byte 2)	16442	23689
STKBOT	16410	23651
STKEND	16512	23653
S TOP	16419	23660
STRLEN	16430	23666
T ADDR	16432	23668
VARS	16400	23627
VERSN	16383	sem equivalência
XPTR	16408	23647

Nem tudo são flores

O que falamos até agora torna fácil a utilização de *alguns* programas desenvolvidos para o TK 90X no TK 85. Veja bem *alguns*.

Um grande número de programas não podem ser traduzidos, principalmente os que fazem amplo uso das caracaterísticas gráficas do 90X. Neste caso, o que podemos fazer é verificar as funções do programa; eliminar as partes gráficas ou adaptá-las

aos recursos do 85. Muitas vezes é mais fácil criar um novo programa, com características semelhantes do que traduzir um programa pronto.

Sempre é possível elaborar bons programas em qualquer computador, desde que saibamos explorar ao máximo suas boas características e respeitar suas limitações, sem esperar que a máquina em questão faça milagres.

Na tabela V mostramos um conjunto de instruções que não podem ser traduzidas para o TK 85 sem a utilização de recursos especiais (periféricos especiais, sub-rotinas em Assembly, alterações no hardware ou na ROM, etc.). Por ela podemos verificar as alterações que não devem ser realizadas em um programa comum.

```
   1 REM ▒1,10,500,80,40,10▒
  10 LET C=16513
  20 GOSUB 9000
  30 PRINT W,
  40 GOTO 20
9000 REM SUB-ROTINA READ
9010 LET W=0
9020 IF PEEK C=26 THEN GOTO 9080
9030 IF PEEK C=234 AND PEEK (C+1
)=8 THEN GOTO 9060
9035 IF PEEK C=118 THEN STOP
9040 LET C=C+1
9050 GOTO 9030
9060 LET C=C+2
9070 LET W=W+PEEK C-28
9080 LET C=C+1
9090 IF PEEK C=8 OR PEEK C=26 TH
EN RETURN
9100 LET W=W*10
9110 GOTO 9070
```

Listagem 1: READ/DATA e RESTORE no TK 85

Estatística, a análise de dados no micro pessoal Parte II

Fabio Augusto Polonio

Na edição anterior da Microhobby, introduzimos alguns conceitos sobre Estatística, demonstrando as técnicas de apresentação de dados por meio de gráficos e tabelas, e citando a importância no mundo atual. Des tacamos, em particular, a Estatística Descritiva, que é a parte desta ciência que se baseia na afirmação de que a "A estatística é coleta, apresentação, análise e interpretação de dados numéricos".

Nosso intuito maior foi situar o leitor neste universo, com exemplos práticos do cotidiano, de maneira que ficasse claro que o objetivo da Estatística é determinar leis de comportamento de uma população, atra vés de um número reduzido de dados.

Dando continuidade aos nossos estudos, este artigo introduzirá conceitos de interpretação e análise de dados. Constituirá o que chamaremos de "análise exploratória dos dados".

Apresentamos um programa aplicativo educacional para ser rodado no TK-2000 além de explorar recursos de interpretação de dados numéricos, que servirá para fixar os conceitos aqui descritos.

Interpretação dos dados

Após a coleta dos dados, eles deverão passar por uma crítica e posteriormente serem arranjados de maneira que fiquem de fácil visualização. A interpretação e análise dos dados coletados nos fornecerão informações relevantes ao caso em estudo.

A crítica dos dados é uma maneira de compilá-los ou resumí-los, tornando-os informativos (obs,diagr,nota rodape), facilitando a comparação com outros dados ou ainda julgando sua adequação à alguma teoria.

Freqüentemente os meios de comunicação visual utilizam gráficos, tabelas e porcentagens para justificar um argumento ou complementar uma explicação e apresentação de um fato.

A interpretação dos dados, considerada uma fase preliminar, fornece informações que constituirão uma fase final, chamada *inferência estatística*. Esta permite que sejam confirmados os dados, testando-se o grau de verdade das informações obtidas.

Tipos de Variáveis

O primeiro conceito, necessário para uma perfeita compreensão do assunto, será o de variável e suas classificações. Comecemos com um exemplo prático:

Um pesquisador da companhia X está interessado em fazer um levantamento sócio-econômico, dos funcionários do departamento administrativo dessa empresa.

Após ter circulado um questionário entre os empregados elaborou a tabela 1.

Para cada elemento investigado tem-se associado um resultado (ou mais), correspondendo à realização de uma certa variável. Consideramos como variável cada um dos campos prescritos na tabela 1 (estado civil, número de filhos, idade, etc.).

Observamos que o pesquisador colheu informações sobre seis variáveis que podem ser divididas em duas classes: variáveis quantitativas e variáveis qualitativas. *Variáveis Quantitativas*: são aquelas que

Tabela 1: Informações sobre estado civil, grau de educação, nro. de filhos, salário (expresso como fração do salário mínimo) idade e procedência de 20 funcionários da companhia X.
Fonte: Dados hipotéticos.

Nro.	Estado Civil	Educação (Grau)	Nro.de Filhos	Salário X.Salário Mínimo	Idade Anos/ Meses	Região de Procedência
1	Solteiro	Primeiro	—	4,00	26 03	Interior
2	Casado	Primeiro	1	4,56	32 10	Capital
3	Solteiro	Segundo	—	5,73	20 10	Outro
4	Solteiro	Primeiro	—	6,66	28 07	Interior
5	Solteiro	Primeiro	—	6,86	42 00	Interior
6	Casado	Segundo	2	7,49	34 10	Capital
7	Casado	Segundo	3	8,12	33 06	Outro
8	Casado	Segundo	2	8,75	27 07	Capital
9	Solteiro	Segundo	—	9,00	30 0l	Interior
10	Solteiro	Segundo	—	9,17	22 02	Capital
11	Casado	Primeiro	3	10,l5	4l 05	Interior
12	Casado	Primeiro	2	10,20	39 08	Interior
13	Solteiro	Superior	—	10,53	25 06	Interior
14	Casado	Segundo	0	11,05	29 08	Outro
15	Solteiro	Segundo	—	11,85	27 08	Interior
16	Solteiro	Superior	—	13,00	26 05	Capital
17	Casado	Superior	1	14,55	30 02	Capital
18	Casado	Segundo	2	16,00	45 04	Outro
19	Casado	Superior	0	17,53	32 0l	Outro
20	Casado	Superior	2	20,33	48 10	Capital

*** Existe uma diferença básica entre dado e informação. Um dado é um fato isolado, enquanto informação é um agregado de fatos organizados ou um dado utilizado com conhecimento e inteligência*

têm como possível realização um número, resultante de uma contagem ou mensuração. Nesta classe de variáveis enquadramos, por exemplo, os campos número de filhos, salário, altura de um indivíduo, peso, etc.

Quando uma variável quantitativa pode assumir, teoricamente, qualquer valor dentro de um intervalo de uma reta real, ela será uma variável *contínua*. Geralmente os valores assumidos por essa variável são resultantes de uma mensuração.

Exemplo: O pêso de um indivíduo é uma variável contínua, pois a rigor, ele pode pesar 60 quilos; 60,5 quilos; 60,57 quilos; etc.

Se os valores assumidos por uma variável quantitativa forem *pontos isolados* de uma reta real, esta será chamada *discreta* e provém, normalmente, de uma contagem. *Exemplo*: número de filhos: 0,1,2,3...

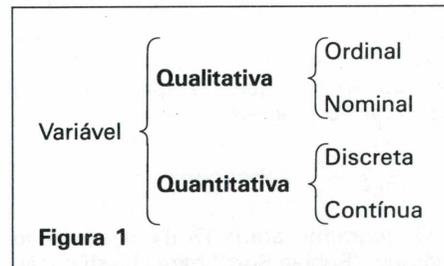
Variáveis qualitativas: quando as possíveis va riações assumidas por uma variável fornecerem uma qualidade ou atributo a um indivíduo pesquisado, ela será uma variável qualitativa.

Ela será qualitativa *ordinal* quando, for possível prever uma certa ordem nos seus valores.

Exemplo: Educação: primeiro grau, segundo grau, superior.

De outra forma, se não há essa possibilidade ela será chamada *nominal. Exemplo*: Região de procedência: interior, Capital, outros...

O esquema da figura 1 resume estes conceitos:

Figura 1

Representação da amostra

Nesta fase, o maior interesse é conhe-

cer as distribuições das possíveis realizações das variáveis. Admite-se que os dados já tenham sido escolhidos, em conformidade com alguma técnica de amostragem.

Esta amostra deve ser representada por tabelas referentes a cada variável, evidenciando sua distribuição de realização e frequência.

Logo que sejam coletados os dados obtemos um cojunto de dados chamados *dados brutos*.

Estes estão na ordem em que foram coletados, sem nenhum arranjo. Assim, o conjunto de dados: 37,49,4l,39,49,35, 40,36,30, constituem os dados brutos.

Após a obtenção desse conjunto, devemos arranjá-lo na ordem crescente ou decrescente de frequência, o que constituirá o "rol". *Exemplo*: 10-10-10-12-12-13-15-17-17-20-23-23-25-26-30-30-30-30-32-33.

A partir do exemplo anterior, fazèmos a distribuição de frequência para a amostra (fig.2).

Xi	Xi
10	3
12	2
13	1
15	1
17	2
20	1
23	2
25	1
26	1
30	4
32	1
33	1

Figura 2

Distribuição de freqüências é o arranjo dos valores e suas respectivas freqüências.

No caso de uma variável discreta, suas possíveis variações são representadas pelos *números absolutos*, que correspondem ao valores assumidos pela variável.

Quando a variável for contínua, ela será representada por intervalos chamados classeo (figura 3).

Figura 3 Exemplo de distribuição de freqüência de variável contínua

Classes	Fi
35! — 45	10
45! — 55	9
55! — 65	23
65! — 75	18
	60

Podemos incrementar a distribuição de frequências com mais dados, como frequência acumulada e relativa.

A frequência acumulada é a soma das frequências dos valores inferiores ou iguais ao valor dado.

Exemplo:

Xi	Fi	F acumulada
0	3	3
1	5	8
2	2	10
	10	

Com a frequência relativa determinamos a porcentagem de um valor na amos tra

Exemplo:

Xi	Fi	F relativa
1	5	5/14
2	7	1/2
3	2	1/7
◇	14	1

Para complementar a representação da amostra, lançaremos mão de recursos gráficos.

O histograma é um gráfico similar ao de barras e representa uma distribuição de frequências por meio de retângulos (no caso de variável contínua) ou retas (variável discreta).

"Através dele podemos obter o polígono de frequência, cujos exemplos podemos observar no programa.

O Programa

O programa Estatis TK II é baseado no método "Bubble Sort" para classificação e arranjo de dados brutos, introduzidos via teclado. Fornece a distribuição de frequências com tabelas e histogramas, para cada tipo de variável (opção do usuário).

No caso de variável contínua, ele calcula o número de classes e as representa na tela com o uso de caracteres gráficos especiais do TK 2000.

Afora esses caracteres especiais, não possui maiores detalhes de digitação.

Na próxima edição incrementaremos o programa com sub-rotinas que o tornem mais completo, fornecendo além da distribuição de frequências, medidas (de posição e dispersão, assunto) que será abordado nesse próximo artigo.

```
10  TEXT
20  REM  ENTRADA DOS DADO
S
30  INPUT "COLOQUE O TAMA
NHO DA AMOSTRA";N
40 T = 1
50  DIM W(N)
60  DIM A(N)
70  FOR I = 1 TO N
80  PRINT "ENTRE COM O ";
I;"' DADO"
90  INPUT A(I)
100  NEXT
110  HOME
120  FOR X = 1 TO N - 1
130  FOR Y = X TO N
140  IF A(X) > A(Y) THEN
 GOSUB 480
150  NEXT : NEXT
160  PRINT "QUAL O TIPO D
E VARIAVEL ?": PRINT "0-DI
SCRETA  1-CONTINUA": GET V
$: IF V$ = "1" THEN 300
170  GR : COLOR  = 5: HLI
N 10,30 AT 1: HLIN 10,30 A
T 6: HLIN 10,30 AT 40
180  FOR V = 10 TO 30 STE
P 10: VLIN 1,40 AT V: NEXT

190  IF V$ = "1" THEN 300

200  HTAB 15: VTAB 3: PRI
NT "XI": HTAB 26: VTAB 3:
PRINT "F"
210 F = 1
220 M = 5
```

ARTIGOS

```
230  FOR I = 1 TO N
240  IF I = N THEN 260
250  IF A(I) = A(I + 1) T
HEN F = F + 1: NEXT
260  HTAB 15: VTAB M: PRI
NT A(I): HTAB 26: VTAB M:
PRINT F:W(T) = F:T = T + 1
:F = 1:M = M + 2
270  IF M ) 20 THEN M = 5
: FOR P = 1 TO 2000: NEXT
: FOR M = 5 TO 20 STEP 2:
HTAB 15: VTAB M: PRINT " "
: HTAB 26: VTAB M: PRINT "
 ": NEXT :M = 5: GOTO 280
280  NEXT
290  GOTO 540
300 K =  INT ( SQR (N) +
.5): IF N (  = 25 THEN K =
 5
310 R = A(N) - A(1)
320 H =  INT ((R / K) + .
5)
330 F = 0
340 M = 6
350 PIVO = A(1)
360  GR : COLOR  = 5: HLI
N 10,30 AT 1: HLIN 10,30 A
T 6: HLIN 10,30 AT 40
370  FOR V = 10 TO 30 STE
P 10: VLIN 1,40 AT V: NEXT

380  FOR I = 1 TO N
390 PAR = PIVO + H
400  IF I = N AND A(I) (
PAR THEN F = F + 1: GOTO 4
30
410  IF A(I) ( PAR THEN F
 = F + 1: GOTO 460
420  HTAB 13: VTAB 3: PRI
NT "CLASSES": HTAB 26: VTA
B 3: PRINT "F"
430  HTAB 14: VTAB M: PRI
NT PIVO;"rcrY";PAR: HTAB 2
6: VTAB M: PRINT F:W(T) =
F:T = T + 1:M = M + 2:F =
1
440  IF M ) 20 THEN M = 6
: FOR P = 1 TO 2000: NEXT
: FOR M = 6 TO 20 STEP 2:
HTAB 12: VTAB M: PRINT "
    ": HTAB 26: VTAB M: PR
INT " ": NEXT :M = 5: GOTO
 280
450 PIVO = PIVO + H
460  NEXT
470  FOR K = 1 TO 2000: N
EXT : GOTO 540
480  REM   ***TROCA DE VA
LORES***
490 AUX = A(X)
500 A(X) = A(Y)
510 A(Y) = AUX
520  RETURN
530  END
540  FOR P = 1 TO 2000: N
EXT : TEXT
550  PRINT " INTERPRETE O
S VALORES DO EIXO X,COMO S
ENDO O NUMERO ORDINAL DA V
ARIAVEL"
560  FOR P = 1 TO 2000: N
EXT
570  HGR : HCOLOR  = 3
580  HPLOT 22,159 TO 250,
159 TO 250,0 TO 22,0
590 B = 10: FOR A = 1 TO
20 STEP 2: VTAB A: PRINT B
: HTAB 4: VTAB A: PRINT "r
b":B = B - 1: HTAB 4: VTAB
 A + 1: PRINT "rZ": NEXT
600 C = 0: FOR A = 4 TO 3
6 STEP 3: HTAB A: VTAB 22:
 PRINT C:C = C + 1: NEXT
610  COLOR  = 3
620 U = 5:G = 7
630  FOR Q = 1 TO T - 1:V
 = 41 - W(Q) * 4
640  FOR P = U TO G: VLIN
 V,39 AT P: NEXT
650 U = U + 3:G = G + 3:
NEXT
```

INÍCIO

N= TAMANHO DA AMOSTRA

A=VARIÁVEL (Matriz unidimensional de N valores)

A(I) REALIZAÇÕES DA VARIÁVEL DADOS BRUTOS

SORT

VARIÁVEL É CONTÍNUA ?

SIM — A

NÃO — B

F= 1 FREQÜÊNCIA

M= 5 LINHA DE IMPRESSORA

1 ≤ I ≤ 200

A(I) = A(I+1) ?

SIM — F= F+1

NÃO — SUB-ROTINA DE IMPRESSÃO

NEXT

A

K= NÚMERO DE CLASSES

R = AMPLITUDE DA AMOSTRA

H = AMPLITUDE DAS CLASSES

F= 0 FREQÜÊNCIA ABS.

PIVO = A(1)

1 ≤ I ≤ N

A(I) < AUX

SIM — C — PIVO;⊢;AUX;F — PIVO = AUX — F = 1 — NEXT

NÃO — D — F= F+1 — NEXT

OBS.: 1

FÓRMULAS USADAS:

$K \overset{N}{=} \sqrt{n}$ para n = número de valores

R = maior dado - menor dado

$H \overset{N}{=}$ R/K para R e K definidos anteriormente.

OBS.: 2

OS CARACTERES GRÁFICOS UTILIZADOS NA LINHA 1060 SÃO OBTIDOS DA MANEIRA:

⟨CONTROL⟩ + ⟨B⟩

⟨SHIFT⟩ + ⟨N⟩

⟨CONTROL⟩ + ⟨SHIFT⟩ + ⟨F⟩

⟨CONTROL⟩ + ⟨B⟩

Super Menus

Cesar de Afonseca e Silva Neto e Wilson José Tucci

Você já deve tê-los utilizado muitas vezes, mas provavelmente poucas vezes ouviu falar deles. Eles são os menus. Um menu se faz necessário toda vez que um programa realiza mais do que uma função, facilitando inclusive, sua escolha.

O menu deve ser original, simpático e atraente, na medida do possível. Podemos dizer inclusive, que um belo menu valoriza ainda mais um bom programa.

O programa apresentado neste número não encerra o assunto em si mesmo, mas apresenta três tipos de menus que, utilizados de maneira original ou incrementados, poderão dar nova vida às apresentações dos seus futuros programas.

O algoritmo utilizado na elaboração dos três menus é praticamente o mesmo. O que muda é a formatação da tela (aqui entra a criatividade do usuário), tornando-os mais ou menos atraentes.

A seguir, analisamos o segundo tipo, que é, sem dúvida, o mais comum: o "Magic Window".

Neste tipo de menu, o posicionamento da opção desejada deve ser feito diretamente pelos números correspondentes ou através das teclas "←" e "→". Feito isto, a escolha pode ser confirmada com a tecla (RETURN). A opção corrente, aquela que está vigorando no momento, será mostrada com uma barra em modo INVERSE a fim de salientá-la. Caso outra seja escolhida, a barra será deslocada para a nova opção (variável OP) e, consequentemente, a opção anterior (variável OV) deverá ser reimpressa em modo NORMAL. Acompanhe a descrição das linhas do programa para melhor compreensão:

LINHA	FUNÇÃO
510	formar a tela base
520	variáveis de controle (veja tabela abaixo)
540	imprime a barra na posição inicial
560 a 630	controle da seleção das opções
640	"apaga" a barra e imprime a opção velha (OV)
650	imprime a barra na nova opção (OP)
660	dá volta ao comando GET

Convém lembrar que as opções que aparecem na tela foram definidas nas linhas 120 a 200, podendo ser mudadas para quaisquer outras.

Bom proveito!!

VARIÁVEL	FUNÇÃO
M$()	vetor das mensagens
OP	opção corrente
OV	opção velha
V,H	posição vertical e horizontal do início do menu
L	número limite de mensagens
DIST	distância entre cada opção no menu
TP$	tipo de menu
P	código ASCII da tecla pressionada

```
100  REM  SUPER MENUS
110  REM  DEFINIR MENSAGE
NS
120 M$(1) = "IMPRIMIR ETI
QUETAS"
130 M$(2) = "PESQUISA ALE
ATORIA"
140 M$(3) = "ALTERAR REGI
STROS"
150 M$(4) = "ENTRAR MATER
IA PRIMA"
160 M$(5) = "CONTROLE DE
ESTOQUE"
170 M$(6) = "LISTAGEM ALF
ABETICA"
180 M$(7) = "EMITIR ANIVE
RSARIOS"
```

```
190 M$(8) = "ATUALISAR RE
GISTROS"
200 M$(9) = "REDEFINIR O
TIPO"
210 OP = 1
220  GOSUB 900
230  VTAB 14: HTAB 5: INP
UT "TIPO (1 A 3) :";TP$
240  ON  VAL (TP$) GOTO 3
00,500,700
300  REM  TIPO #1
310  GOSUB 900
320  VTAB 6: HTAB 17: PRI
NT "TIPO";: INVERSE : PRIN
T TP$: NORMAL
330 H = 9:V = 8:L = 9:DIS
T = 1
340  FOR I = 1 TO L: VTAB
 V + I * DIST: HTAB H: PRI
NT I; SPC( 2);M$(I): NEXT

350  VTAB V + OP * DIST:
HTAB H - 4: PRINT "--> ";O
P; SPC( 2);: INVERSE : PRI
NT M$(OP): NORMAL
360  GET P$:P =  ASC (P$)

370  IF P = 13 THEN 460
380  IF P = 21 THEN OV =
OP:OP = OP + 1
390  IF P = 8 THEN OV = O
P:OP = OP - 1
```

```
400  IF P > = 49 AND P <
 = 49 + L THEN OV = OP:OP
 = P - 48
410  IF OP < 1 THEN OP =
L
420  IF OP > L THEN OP =
1
430  VTAB V + OV * DIST:
HTAB H - 4: PRINT  SPC( 4)
;OV; SPC( 2);M$(OV)
440  VTAB V + OP * DIST:
HTAB H - 4: PRINT "--> ";O
P; SPC( 2);: INVERSE : PRI
NT M$(OP): NORMAL
450  GOTO 360
460  IF OP = L THEN 220
470  GOSUB 960: GOTO 300
500  REM  TIPO#2
510  HOME : GOSUB 900
520 V = 8:H = 11:L = 9:DI
ST = 1
530  VTAB 6: HTAB 17: PRI
NT "TIPO";: INVERSE : PRIN
T TP$: NORMAL
540  FOR I = 1 TO L: VTAB
 V + I * DIST: HTAB H: PRI
NT I; SPC( 1);M$(I): NEXT

550  INVERSE : VTAB V + O
P * DIST: HTAB 4: PRINT  S
PC( H - 4);OP; SPC( 1);M$(
OP); TAB( 38): NORMAL
560  GET P$
570 P =  ASC (P$)
580  IF P = 13 THEN 670
590  IF P = 21 THEN OV =
OP:OP = OP + 1
600  IF P = 8 THEN OV = O
P:OP = OP - 1
610  IF P > = 49 AND P <
 = 49 + L THEN OV = OP:OP
 = P - 48
620  IF OP < 1 THEN OP =
L
630  IF OP > L THEN OP =
1
640  VTAB V + OV * DIST:
HTAB 4: PRINT  SPC( H - 4)
;OV; SPC( 1);M$(OV); TAB(
38)
650  INVERSE : VTAB V + O
P * DIST: HTAB 4: PRINT  S
PC( H - 4);OP; SPC( 1);M$(
OP); TAB( 38): NORMAL
660  GOTO 560
670  IF OP = L THEN 220
680  GOSUB 960: GOTO 500
700  REM  TIPO#3
710  GOSUB 900
720  VTAB 6: HTAB 17: PRI
NT "TIPO";: INVERSE : PRIN
T TP$: NORMAL
730 H = 10:V = 8:L = 9:DI
ST = 1
740  FOR I = 1 TO L: VTAB
 V + I * DIST: HTAB H: PRI
NT I; SPC( 2);M$(I): NEXT

750  VTAB V + OP * DIST:
HTAB H - 1: PRINT  CHR$ (9
1);OP;"^"; SPC( 1);M$(OP)
760  VTAB 20: HTAB 12: IN
VERSE : PRINT M$(OP);: NOR
MAL : PRINT  TAB( 39)
770  GET P$
780 P =  ASC (P$)
790  IF P = 13 THEN 885
800  IF P = 21 THEN OV =
OP:OP = OP + 1
810  IF P = 8 THEN OV = O
P:OP = OP - 1
820  IF P > = 49 AND P <
 = 49 + L THEN OV = OP:OP
 = P - 48
830  IF OP < 1 THEN OP =
L
840  IF OP > L THEN OP =
1
850  VTAB V + OV * DIST:
HTAB H - 1: PRINT  SPC( 1)
;OV; SPC( 2);M$(OV)
860  VTAB V + OP * DIST:
HTAB H - 1: PRINT  CHR$ (9
1);OP;"^"; SPC( 1);M$(OP)
870  VTAB 20: HTAB 12: IN
VERSE : PRINT M$(OP);: NOR
MAL : PRINT  TAB( 39)
880  GOTO 770
885  IF OP = L THEN 220
890  GOSUB 960: GOTO 700
900  REM  MONTAR A TELA
910  HOME : INVERSE
920  VTAB 2: PRINT ":"; S
PC( 11): HTAB 28: PRINT  S
PC( 12);":": VTAB 23: PRIN
T ":"; SPC( 38);":";
930  FOR I = 3 TO 22: VTA
B I: HTAB 1: PRINT " ";: H
TAB 40: PRINT " ";: NEXT
940  VTAB 1: HTAB 14: PRI
NT ":"; SPC( 11);":": HTAB
 14: PRINT " SUPER MENUS "
: HTAB 14: PRINT ":"; SPC(
 11);":": NORMAL
950  RETURN
960  REM  SAIDA DOS MENUS

970  HOME : VTAB 14: HTAB
 10: PRINT M$(OP): NORMAL
: FOR I = 1 TO 1000: NEXT
: RETURN
```

Integração Númerica no TK 2000

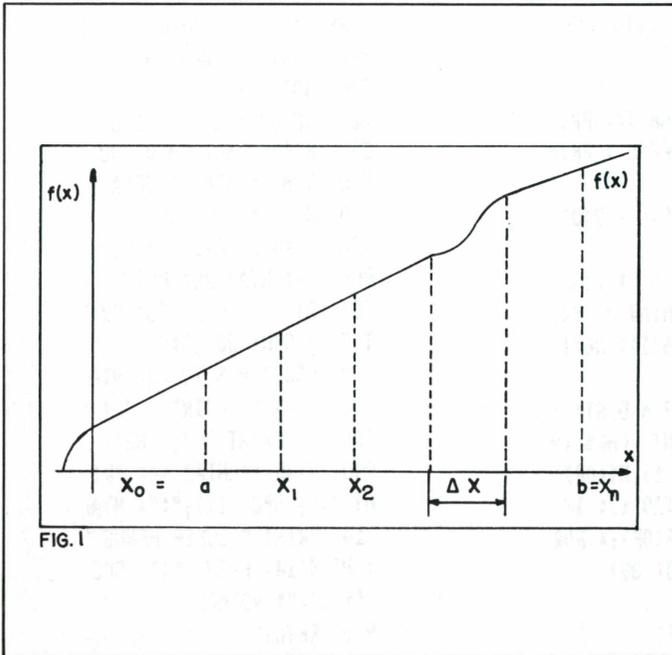
Ralph Marques Aguiar

O principal objetivo deste programa desenvolvido no TK-2000 é calcular a integral definida entre dois pontos, usando para isto, três métodos de integração numérica.

A função pode ser apresentada de duas formas:

Analítica:Y=(x), a função é definida através de uma equação.

Numérica: a função é definida através do seu domínio e da sua imagem, exemplo na tabela A.

i	0	1	2	3	4	5	6	7	8	
Xi	0	0,5	1,0	1,5	2	2,5	3,0	3,5	4	→ DOMINIO
Fi	1,5	0,75	0,5	0,75	1,5	2,75	4,0	6,75	10	→ IMAGEM

Exemplo de integração numérica
Exemplo de integração numérica usando a fórmula dos retângulos

Suponhamos que no intervalo [a,b] seja dada uma função Y=f(x) contínua e deseja-se calcular a $\int_a^b f(x)dx$.

Vamos subdividir o intervalo [a,b] nos pontos

$a = X_0, X_1, X_2, X_3, ..., X_n = b$

em "n" partes iguais, de comprimento igual, H=(letra delta)x, onde:

$$\Delta x = \frac{b - a}{n}$$

Representemos por $Y_0, Y_1, Y_2, Y_3, ..., Y_{n-1}, Y_n$ os valores da função f(x) em $X_0, X_1, X_2, X_3, ..., X_{n-1}, X_n$, isto é:

$Y_0 = f(x_0)$
$Y_1 = f(x_1)$
$Y_2 = f(x_2)$

$Y_n = f(x_n)$

Façamos as somas:

$Y_0 \Delta x + Y_1 \Delta x + Y_2 \Delta x + ... + Y_{n-1} \Delta x Y_1 \Delta x + Y_2 \Delta x + Y_3 \Delta x + ... + Y_n \Delta$

Cada uma destas somas é a "Soma Integral" para f(x) no intervalo [a,b]. Por isso, representa aproximadamente a integral:

$$\int_a^b f(x)dx = \frac{b-a}{n} (Y_0 + Y_1 + Y_2 + ... + Y_{n-1})$$

$$\int_a^b f(x)dx = \frac{b-a}{n} (Y_0 + Y_1 + Y_2 + Y_3 + ... + Y_n)$$

As expressões 1 e 2 são fórmulas dos retângulos. Na figura I, torna-se claro que, se f(x) é uma função positiva e crescente, então a expressão 1 representa a área da figura, composta pelos retângulos abaixo da curva Y=f(x) (retângulos inferiores) e a expressão 2 fornece a área da figura composta pelos retângulos superiores.

Os erros cometidos no cálculo de integrais pela aplição da fórmula dos retângulos, será tanto menor quanto maior for o número n (isto é, quanto menor for o passo H=Δx).

Procedimentos de execução:

Devemos primeiramente definir o tipo (ou forma) da função da qual iremos calcular a integral.

Se a forma for *analítica*, a função passará para a forma BASIC (operadores aritméticos da Linguagem) através do comando:

450 DEF FN F(x) = XXX

→ Indicação do cursor na tela → Função na forma BASIC

Após definir a função, digite RUN 2000, que retorna à execução do programa. A seguir será executada a função em questão, para os valores de x no intervalo de [2,10]. No caso dos valores encontrados para Y sejam os esperados prossiga com o processamento. Caso contrário, a opção *não* retorna ao processo, para o ponto de entrada de f(x).

Os valores de x e y mostrados no teste de [2,10], terão apresentadas apenas as quatro primeiras casas dos números.

Exemplo: 153878 → 1538 será o valor mostrado.

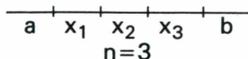
Se a forma for numérica, a imagem da função será passada ao programa como o indicado no processamento. Observe que somente a imagem será passada.

Do domínio serão passados **a** e **b**.

Domínio e imagem já foram mostrados na tabela A.

Em ambas as formas acima deve-se passar ao programa o número de intervalos **N** e os limites inferior e superior para cálculo de $\Delta x = \frac{b-a}{n}$ fundamental ao desenvolvimento de ambas as desenvolvimento de ambas as formas e todos os métodos.

Para a forma numérica: $N=(i_n - i_1) - 1$, como exemplo, na tabela A: $N=(8-0)-1=7$

Para a forma analítica: N=n. Tendo em vista que:

Observação:

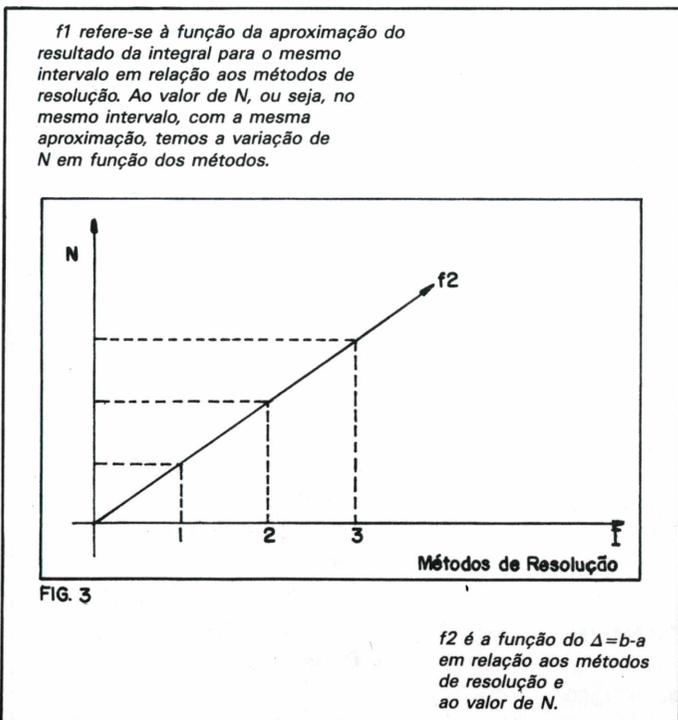
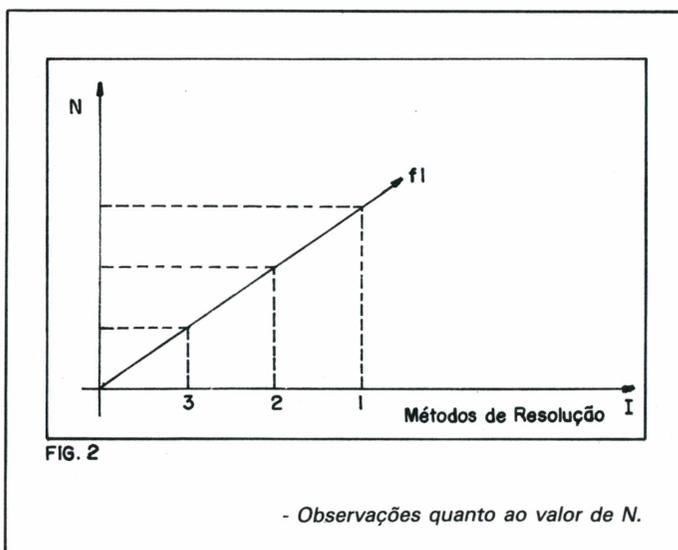
N nunca pode ser maior que 400 a e b devem satisfazer a condição a<b

FIG. 2

- Observações quanto ao valor de N.

f1 refere-se à função da aproximação do resultado da integral para o mesmo intervalo em relação aos métodos de resolução. Ao valor de N, ou seja, no mesmo intervalo, com a mesma aproximação, temos a variação de N em função dos métodos.

FIG. 3

f2 é a função do Δ=b-a em relação aos métodos de resolução e ao valor de N.

```
10  REM  ****************
***********
11  REM  *         INTEGR
AL         *
12  REM  *        RIO 15/0
8/85       *
13  REM  *AUTOR:RALPH MAR
QUES AGUIAR*
14  REM  *OBJETIVO: EFETU
AR CALCULOS*
15  REM  * DE INTEGRAIS P
OR  METODO *
16  REM  *         NUMERI
CO         *
17  REM  ****************
***********
18  REM
20  SPEED= 50
25  CLEAR
30  HOME : VTAB 3: HTAB 1
3: PRINT "I N T E G R A L"

40  VTAB 6: HTAB 5: PRINT
 "AUTOR : RALPH MARQUES AG
UIAR"
50  VTAB 8: HTAB 5: PRINT
 "VERSAO 1.3"
55  VTAB 10: HTAB 5: PRIN
T "RIO, AGOSTO DE 1985"
60  SPEED= 255
70  HOME : VTAB 2: HTAB 1
4: PRINT "MENU PRINCIPAL"
80  VTAB 6: HTAB 4: PRINT
 "ESTE PROGRAMA CALCULA A
INTEGRAL DE"
90  PRINT  TAB( 4)"UMA FU
NCAO DA FORMA ANALITICA Y=
F(X)"
100  PRINT  TAB( 4)"OU SO
B A FORMA NUMERICA."
110  PRINT  TAB( 4)"RESPO
NDA A SEGUIR QUAL A FORMA"

120  PRINT  TAB( 4)"DE SU
A FUNCAO."
125  PRINT " "
130  PRINT  TAB( 7)"OBS:
OUTRO MENU A SEGUIR INDICA
RA"
140  PRINT  TAB( 12)"QUAL
 O METODO DESEJADO"
150  VTAB 15: HTAB 6: PRI
NT "(1) FORMA ANALITICA"
160  VTAB 17: HTAB 6: PRI
NT "(2) FORMA NUMERICA"
170  VTAB 19: INPUT "ENTR
E COM 1 OU 2 ==> ";TP%
180  IF TP% = 1 OR TP% =
2 THEN  GOTO 200
190  SOUND 100,100 TO 100
,100
192  PRINT " ": PRINT  TA
B( 3)"ERRO NA OPCAO --> RE
DIGITE"
196  GOTO 170
200  ON TP% GOSUB 330,610

210  GOSUB 690
220  ON MT% GOSUB 810,870
,930
230  HOME : VTAB 3: HTAB
8: PRINT "--VALOR DA INTEG
RAL--"
240  VTAB 9: PRINT  TAB(
11)"I= " TAB( 15)AD
250  VTAB 18: HTAB 7: PRI
NT "DESEJA CALCULAR NOVA I
NTEGRAL?"
255  PRINT " "
260  INPUT "RESPONDA (S/N
) ==> ";R$
270  IF R$ <  > "S" THEN
 GOTO 280
274  CLEAR
276  GOTO 70
280  END
330  REM  ***************
**
332  REM  *FORMA ANALITIC
A*
334  REM  ***************
**
340  HOME : VTAB 3: HTAB
12: PRINT "FORMA ANALITICA
"
350  VTAB 6: HTAB 5: PRIN
T "DIGITE A FUNCAO NA FORM
A BASIC"
355  VTAB 7: HTAB 5: PRIN
T "COMO A SEGUIR:"
360  VTAB 9: HTAB 7: PRIN
T ">450 DEF FN F(X)=XX OND
E XX=FUNCAO"
370  VTAB 11: HTAB 5: PRI
NT "APOS DIGITAR F(X), COM
```

```
ANDAR:"
380  VTAB 13: HTAB 7: PRI
NT "RUN 2000 --> RETORNO A
 EXECUCAO"
390  VTAB 16: HTAB 2: PRI
NT "OBS: CASO HAJA ERRO DE
 NUMERACAO"
400  VTAB 17: HTAB 7: PRI
NT "RESSETAR O PROGRAMA E
REINICIA-LO"
410  END : RETURN
420  REM  *RETOMA PROCESS
O*
430  HOME : VTAB 2: HTAB
7: PRINT "TESTE DA FUNCAO
DE 2 A 10"
440  VTAB 5: HTAB 13: PRI
NT "    !  Y  "
445  VTAB 6: HTAB 13: PRI
NT "--------------------"

448  PRINT " "
450  DEF  FN F(X) = 20 ^
2
455  FOR X = 2 TO 10
460 Y =  FN F(X)
470 E$ =  STR$ (X):D$ =
STR$ (Y)
475 EX$ =  MID$ (E$,1,4):
DY$ =  MID$ (D$,1,4)
480  PRINT  TAB( 14)EX$ T
AB( 19)"!" TAB( 21)DY$
490  NEXT X:X = 0
500  PRINT " ": PRINT  TA
B( 5)"A FUNCAO ESTA OK?"
510  PRINT " ": INPUT "RE
SPONDA (S/N) => ";T$
520  IF T$ = "S" THEN  GO
TO 530
525  GOSUB 330
530 TP% = 1: GOSUB 1010:
REM  ENTRADA DE A,B,N
540  REM  *CALCULO DOS VA
LORES DE F(X)*
550 IN% = N + 1:IV = A
555  FOR I = 1 TO IN%
560 X = IV
565 CY(I) =  FN F(X)
570 IV = IV + H
580  NEXT I
590 LT% = IN%
600  GOTO 210
610  REM  **************
*
614  REM  *FORMA NUMERICA
*
616  REM  **************
*
620  GOSUB 1010: REM  ENT
RADA DE A,B,N
625 TP% = 2
630  HOME : VTAB 3: HTAB
13: PRINT "FORMA NUMERICA"
640  VTAB 6: HTAB 5: PRIN
T "DIGITE AGORA OS VALORES
 DE F(X)"
646 IN% = N + 2
650  FOR I = 1 TO IN%
660  PRINT " ": INPUT "EN
TRE F(X) ==> ";CY(I)
670  NEXT I
675 LT% = IN%
680  RETURN
690  REM  **************
*****
700  REM  *ROTINA TELA ME
TODO*
705  REM  **************
*****
708  HOME : VTAB 3: HTAB
5: PRINT "ESCOLHA O METODO
 DE RESOLUCAO"
710  VTAB 5: HTAB 5: PRIN
T "ENTRE COM SUA OPCAO:"
720  VTAB 8: HTAB 9: PRIN
T "(1) FORMULA DOS RETANGU
LOS"
730  VTAB 10: HTAB 9: PRI
NT "(2) FORMULA DOS TRAPEZ
IOS"
740  VTAB 12: HTAB 9: PRI
NT "(3) FORMULA DE SIMPSON
 1/3"
750  VTAB 15: INPUT "ENTR
E COM O NUMERO DA OPCAO ==
> ";MT%
760  IF MT% < 4 AND MT% >
 0 THEN  GOTO 800
770  SOUND 100,100 TO 100
,100
780  PRINT " ": PRINT "**
*ERRO DE OPCAO***"
790  GOTO 750
800  RETURN
810  REM  **************
***
812  REM  *ROTINA RETANGU
LO*
814  REM  **************
***
820 S = 0:AD = 0
830  FOR I = 1 TO LT%
840 S = S + CY(I)
842  NEXT I
850 AD = H * S
860  RETURN
870  REM  **************
**
872  REM  *ROTINA TRAPEZI
O*
874  REM  **************
**
880 S = 0:AD = 0:TR% = LT
% - 1
890  FOR I = 2 TO TR%
900 S = S + CY(I)
906  NEXT I
910 RL = (CY(1) + CY(LT%)
) / 2
912 AD = H * (RL + S)
920  RETURN
930  REM  **************
*****
932  REM  *ROTINA SIMPSON
 1/3*
936  REM  **************
*****
940 IM% = LT% - 1:PR% = L
T% - 2:SI = 0:SP = 0
950  FOR K = 2 TO IM% STE
P 2
960 SI = SI + CY(K)
962  NEXT K
970  FOR I = 3 TO PR% STE
P 2
980 SP = SP + CY(I)
982  NEXT I
990 AD = (H / 3) * (CY(1)
 + CY(LT%) + (4 * SI) + (2
 * SP))
1000  RETURN
1010  REM  *************
*******
1012  REM  *ROTINA ENTRAD
A A,B,N*
1014  REM  *************
*******
1020  HOME : VTAB 3: HTAB
 5: PRINT "FORNECA O NUMER
O DE INTERVALOS"
```

```
1030  VTAB 6: INPUT "ENTR
E NUMERO ==> ";N
1032  IF N < 401 AND N >
0 THEN  GOTO 1040
1034  SOUND 100,100 TO 10
0,100
1036  VTAB 8: HTAB 2: PRI
NT "ERRO --> N=0 OU N>400"

1038  GOTO 1030
1040  DIM CY(402)
1050  VTAB 10: HTAB 5: PR
INT "FORNECA OS INTERVALOS
 DE INTEGRACAO"
1060  VTAB 12: INPUT "ENT
RE LIMITE INFERIOR ==> ";A

1070  VTAB 14: INPUT "ENT
RE LIMITE SUPERIOR ==> ";B

1080  IF (B > A) THEN  GO
TO 1120
1090  SOUND 100,100 TO 10
0,100
1100  VTAB 16: HTAB 2: PR
INT "ERRO --> LIM.INF>=LIM
.SUP"
1110  GOTO 1060
1120  IF TP% = 1 THEN AUX%
 = N
1125  IF TP% = 2 THEN AUX%
 = N + 1
1127 H = (B - A) / AUX%
1130  RETURN
1999  REM  *RETORNO AO PR
OCESSO*
2000  GOTO 420
```

O autor é programador pleno da Rede Ferroviaria Federal S.A.

Explorando o Teclado do TK 2000

Alvaro A.L.Domingues

Use em seus programas, de forma inteligente, os caracteres gráficos especiais e os PEEKs de leitura do teclado de seu micro

O teclado do TK-2000 possui, além dos números, letras e símbolos especiais, uma série de caracteres gráficos que podem ser usados para melhorar telas gráficas ou de textos, ajudando dessa forma, na criação de jogos ou aplicativos e na simulação da função INKEY$ através do uso da função PEEK (39). Estes dois últimos recursos serão explicados a fundo, visando eliminar dúvidas e mostrar outras possibilidades do TK 2000.

Qual a tecla que foi pressionada?

Dentre os recursos normais do TK-2000 existe a instrução GET, cuja função é esperar que o usuário digite uma tecla qualquer, que será armazenada posteriormente, numa variável alfanumérica. O processamento do programa nesta fase é então interrompido, continuando apenas após a digitação de alguma tecla (em certos programas como processadores de texto e jogos esta interrupção não é interessante, podendo até ser prejudicial para algumas funções).

Alguns microcomputadores, como por exemplo o TK 90X, o TK-85 e o TRS-80, não possuem a instrução GET. Estas máquinas têm a função INKEY$ que efetua a leitura das teclas, à medida que vão sendo pressionadas, sem parar o processamento. Se, nenhuma tecla for apertada o micro passa para a instrução seguinte.

A solução, para se simular o INKEY$, no caso do TK-2000, é utilizar-se do PEEK (39). Todavia, existe um problema: o código obtido para cada tecla é diferente do código ASCII. O que o usuário pode fazer é utilizar o seguinte programa:

```
10 PRINT PEEK (39)
20 GOTO 10
```

A partir dele poderá, digitando tecla por tecla, obter todos os códigos do teclado.

Ao lado desta instrução está o PEEK (38), que fornece um valor diferente para determinados grupos de teclas, de acordo com a zona do teclado a que pertencem. Na tabela I apresentamos todos os códigos do PEEK (38) e do PEEK (39) e na figura 1 mostramos as zonas do teclado abrangidas pelo PEEK (38)

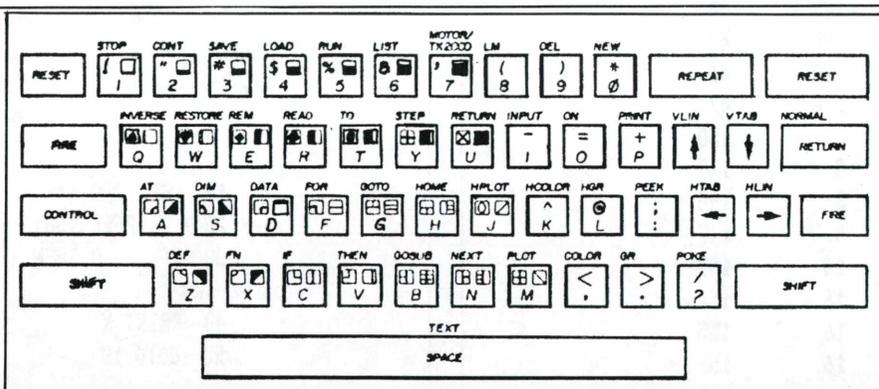



FIG. 1

Tabela I - PEEK (38) e PEEK (39)

CARACTERE	PEEK(38)	PEEK(39)
NENHUM	0	48
RETURN	128	42
1	8	23
2	8	22
3	8	21
4	8	20
5	8	19
6	16	25
7	16	26
8	16	27
9	16	28
0	16	29
Q	4	17
W	4	16
E	4	15
R	4	14
T	4	13
Y	32	31
U	32	32
I	32	33
O	32	34
P	32	35
A	2	11
S	2	10
D	2	9
F	2	8
G	2	7
H	64	37
J	64	38
K	64	39
L	64	40
:	64	41
Z	1	5
X	1	4
C	1	3
V	1	2
B	1	1
N	128	43
M	128	44
,	128	45
.	128	46
?	128	47
"	8	150
#	8	149
$	8	148
%	8	147
&	16	153
'	16	154
(	16	155
)	16	156
*	16	157
-	32	161
=	32	162
+	32	163
^	64	167
@	64	168
;	64	169
<	128	173
>	128	174
/	128	175
SETAS		
↑	64	36
↓	32	30
←	8	18
→	16	24

Uma vez que o número de cada tecla é diferente e eles não estão "jogados" ao acaso, é possível convertê-los para o código ASCII. Assim, poderemos desenvolver uma sub-rotina em BASIC ou Assembly que efetua a tradução. Os programas que mostraremos a seguir fazem esta transformação. O primeiro deles (listagem 1) é em BASIC e o segundo é em Assembly (listagem 2).

```
10 X = PEEK (39):Y = IN
T (X / 64):A = X
20 IF Y > 0 THEN A = X -
 Y * 64
30 CD = PEEK (62880 + A
+ Y + 3 * (A - 1)) - 128
40 IF CD < 0 THEN CD = C
D + 128
50 PRINT CD: GOTO 10
```

Listagem 1 - Programa em BASIC para leitura do teclado.

Listagem 2 - Programa para leitura do teclado em Assembly.

```
0300-   20 43 F0     JSR  $F043
0303-   85 3A        STA  $3A
0305-   60           RTS
```

a) Assembly

```
10 CALL 768
20 X = PEEK (58) - 128
30 IF X < 0 THEN X = X +
 128
40 PRINT X
50 GOTO 10
```

b) BASIC (para teste).

Os caracteres especiais

Uma das coisas que torna o TK 2000 diferente do Apple é a existência de uma série de caracteres gráficos especiais, que podem ser usados no modo texto. Isso possibilita a criação de uma variada gama de telas. Além disso, podemos misturar telas em alta e baixa-resolução com caracteres de texto, sem termos que abrir "janelas" como ocorre com o Apple.

Estes caracteres são obtidos após digitar-se CONTROL-B, ou após um PRINT CHR$ (242), passando o teclado para o modo gráfico. Como podemos observar na figura 5, a grande maioria das teclas possui, além dos caracteres normais, um ou dois caracteres gráficos. Após ser dado um CONTROL-B digitamos simultaneamente SHIFT+TECLA ou CONTROL+SHIFT+TECLA. Por exemplo, SHIFT-Q após um CONTROL-B, mostra o naipe espadas na tela. Para voltarmos ao normal, devemos digitar novamente CONTROL-B. Podemos usar isso numa linha de programa, da seguinte forma:

a) digitando a instrução (PRINT ou LET A$=, por exemplo);

b) abrindo aspas;

c) digitando os caracteres que comporão a figura (letras, números, símbolos especiais ou caracteres gráficos);

d) digitando CONTROL-B;

e) fechando aspas.

Observe a figura 2. Ela mostra o teclado com todos os símbolos gráficos. O símbolo gráfico da esquerda de cada tecla é obtido digitando-se SHIFT+TECLA. O símbolo gráfico da direita de cada tecla é obtido por CONTROL+SHIFT+TECLA.

Entretanto o usuário pode sentir dificuldade em listar, em impressora, estes símbolos especiais, visto que seus códigos correspondem a outros caracteres da mesma.

Para evitar esta dificuldade, mostramos na tabela II o que aparece numa listagem impressa quando usamos caracteres gráficos especiais, com a indicação de como obtê-los.

Mais detalhes sobre o uso do teclado podem ser obtidos no Manual de Operações do TK 2000, nas páginas de 22 a 27.

Explorando mais o TK 2000

Estas tabelas que publicamos permitem a você elaborar programas mais versáteis em seu TK 2000. O uso do teclado em todo o seu potencial,possibilita o desenvolvimento de sua criatividade, sobretudo na criação de aplicativos gráficos e jogos, onde a manipulação de telas por meio de teclas é fundamental e o uso de símbolos gráficos é inerente ao processo.

Tabela II - Como os caracteres especiais aparecem na impressora

Código	Impressora	Teclado
193	rA	SHIFT+CONTROL+1
194	rB	SHIFT+CONTROL+2
195	rC	SHIFT+CONTROL+3
196	rD	SHIFT+CONTROL+4
197	rE	SHIFT+CONTROL+5
198	rF	SHIFT+CONTROL+6
199	rG	SHIFT+CONTROL+7
200	rH	SHIFT+CONTROL+Q
201	rI	SHIFT+CONTROL+W
202	rJ	SHIFT+CONTROL+E
203	rK	SHIFT+CONTROL+R
204	rL	SHIFT+CONTROL+T
205	rM	SHIFT+CONTROL+Y
206	rN	SHIFT+CONTROL+U
207	rO	SHIFT+CONTROL+I
208	rP	SHIFT+CONTROL+G
209	rQ	SHIFT+CONTROL+H
210	rR	SHIFT+CONTROL+B
211	rS	SHIFT+CONTROL+N
212	rT	SHIFT+CONTROL+A
213	rU	SHIFT+CONTROL+S
214	rV	SHIFT+CONTROL+Z
215	rW	SHIFT+CONTROL+X
216	rX	SHIFT+CONTROL+D
217	rY	SHIFT+CONTROL+F
218	rZ	SHIFT+CONTROL+C
219	r[	SHIFT+CONTROL+V
220	r\	SHIFT+CONTROL+J
221	r]	SHIFT+M
222	r^	SHIFT+T
223	r-	SHIFT+J
224	r'	SHIFT+G
225	ra	SHIFT+H
226	rb	SHIFT+B
227	rc	SHIFT+N
228	rd	SHIFT+Q
229	re	SHIFT+W
230	rf	SHIFT+E
231	rg	SHIFT+R
232	rh	SHIFT+D
233	ri	SHIFT+F
234	rj	SHIFT+C
235	rk	SHIFT+A
236	rl	SHIFT+S
237	rm	SHIFT+V
238	rn	SHIFT+Z
239	ro	SHIFT+X
240	rp	SHIFT+U
241	rq	SHIFT+Y
242	rr	SHIFT+M

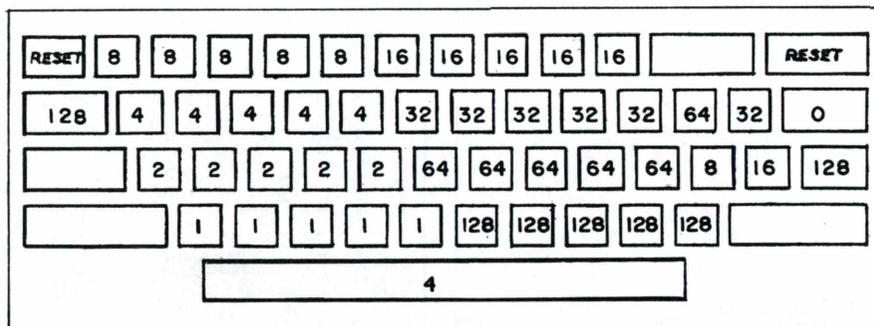



FIG.2

Tabuada

Aprendendo a Tabuada com o micro TK 90X

Atenção garotada do curso primário que tem tido dificuldades em decorar a tabuada. Este programa lhes ajudará a aprendê-la, ao mesmo tempo que lhes ensinará como mexer com o micro.

Operando a tabuada

Ao rodar o programa, o micro lhe perguntará qual tabuada será usada. Em seguida, duas opções lhe serão apresentadas para que você escolha entre multiplicação e divisão, com operações de 1 a 12.

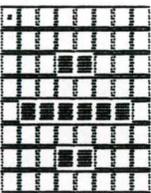
Em ambas as operações, você terá que resolver dez cálculos distintos, escolhidos aleatoriamente. Se você conseguir resolver todos os cálculos sem cometer nenhum erro, ao final, o computador lhe mostrará uma mensagem parabenizando-o.

O programa oferece também, ao final, as opções de troca da tabuada ou repetição da existente. Durante a realização da operação, se você entrar com o resultado errado, por duas vezes consecutivas, o micro se encarregará de fazer a correção.

No programa está definida a execução da tabuada do número 12. Mas, você pode modificar, aumentando ou diminuindo este valor, apenas alterando as linhas 45 e 280 do programa Tabuada (listagem 1).

A função UDG é utilizada no programa, porém, apenas para definir o símbolo de divisão, como mostra a tabela 1.

"Tabuada" despertará o interesse da moçada, que está iniciando a carreira escolar, estimulando-a a decorar via computador, ao mesmo tempo familiarizando-a com o novo companheiro na escola.

Tabela 1.

Listagem 1

```
   1 REM ** TABUADA **
  10 GOSUB 2000
  20 CLS
  30 PRINT "Escolha uma Tabuada?
"'"(Entre de 1 a 12)"
  40 INPUT b
  45 IF b>12 THEN GOTO 20
  50 PRINT ;b;
  60 PRINT
  70 PRINT "Voce quer Multiplica
r ou Dividir?"'"(Entre com m ou
d)"
  80 INPUT x$
  90 PRINT ;x$;
 100 LET f=0
 110 LET g=0
 120 PAUSE 50: GOTO 150
 130 PRINT "Correto": IF cont=0
THEN LET f=f+1
 140 PAUSE 100
 150 CLS
 160 IF g=10 AND f=10 THEN PRINT
 AT 10,2; FLASH 1;"Parabens, Voc
e realizou todos   os calculos s
em cometer erros"; FLASH 0: PRIN
T AT 15,2,"Pressione""r"" para r
epitir ou ""t"" para mudar de ta
buada": GOTO 190
 170 IF g=10 AND f<>10 THEN PRIN
T AT 10,2; FLASH 1;"Voce ultrapa
ssou 10 Operacoes"; FLASH 0: PRI
NT AT 15,2;"Pressione ""r"" para
 repitir ou ""t"" para mudar de
tabuada": GOTO 190
 180 GOTO 240
 190 IF INKEY$="" THEN GOTO 190
 200 LET q$=INKEY$
 210 IF q$="t" THEN GOTO 20
 220 IF q$="r" THEN GOTO 100
 230 IF q$<>"t" AND q$<>"r" THEN
 GOTO 190
 240 CLS
 250 PRINT AT 20,10;"Pontuacao "
;f: PRINT AT 21,10;"Sobre ";g
 260 LET d=0
 270 LET cont=0
 280 LET a=1+INT (RND*12)
 290 LET h=a*b
 300 IF x$="m" THEN GOTO 320
 310 IF x$="d" THEN GOTO 410
 320 PRINT AT 0,0;a;"x";b;"=?"
 330 LET c=h
 340 GOTO 900
 350 LET cont=cont+1
 360 IF x$="m" AND cont=2 THEN P
RINT "Errado, ";a;"x";b;"=";h: P
AUSE 200: GOTO 140
```

```
 370 IF x$="d" AND cont=2 THEN P
RINT "Errado, ";h;"÷";b;"=";a;:
PAUSE 200: GOTO 140
 380 LET d=2
 390 PRINT "Errado, Tente Novame
nte": PAUSE 100
 400 GOTO 1000
 410 PRINT AT 0,0;h;"÷";b;"=?"
 420 LET c=a
 900 LET g=g+1
1000 IF INKEY$<>"" THEN GOTO 100
0
1010 IF INKEY$="" THEN GOTO 1010
1020 LET k=VAL INKEY$
1030 PRINT AT 1+d,0;k
1040 IF k=c THEN PAUSE 100: GOTO
 130
1050 IF k>c THEN GOTO 350
1060 IF c=10 AND k<10 AND k<>1 T
HEN GOTO 350
1070 IF k<c AND c-10<0 THEN GOTO
 350
1090 IF INKEY$<>"" THEN GOTO 109
0
1100 IF INKEY$="" THEN GOTO 1100
1110 LET l=VAL INKEY$
1120 LET k=k÷10+l
1130 PRINT AT 1+d,0;k
1140 IF k=c THEN PAUSE 100: GOTO
 130
1150 IF x>c THEN GOTO 350
1160 IF c=100 AND x<100 AND x>10
 THEN GOTO 350
1170 IF x<c AND c-100<0 THEN GOT
O 350
1180 IF INKEY$<>"" THEN GOTO 118
0
1190 IF INKEY$="" THEN GOTO 1190
1200 LET o=VAL INKEY$
1210 LET k=k÷10+o
1220 PRINT AT 1+o,0;k
1230 IF k=c THEN PAUSE 100: GOTO
 130
1240 IF k<>c THEN GOTO 350
2000 FOR n=0 TO 7
2010 READ x
2020 POKE USR "÷"+n,x
2030 NEXT n
2040 DATA 0,0,24,0,126,0,24,0
2050 RETURN
```

Reúna a turma para uma partida de Boliche com o TK 90X

Que tal reunir a turma do sábado à noite para um jogo de boliche?...

Há algum tempo atrás, o boliche não era muito difundido entre os brasileiros, mas essa imagem mudou e hoje em dia já existem inúmeros adeptos desse esporte.

Equipes foram formadas, e se criaram campeonatos que são disputados de uma forma séria, onde todos dão o máximo de sí para conseguir boas colocações.

Antigamente, a contagem de pontos era feita de forma simples, praticamente à mão. Hoje temos apenas o trabalho de pegar e lançar a bola e o computador se encarrega do resto. Cada pista de jogo possui uma tela onde os pontos são registrados, e o jogador tem diante de seus olhos todas as suas pontuações, jogada por jogada.

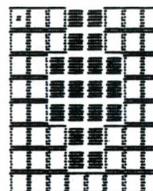
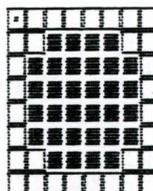
Mas, se você não está a fim de sair com o pessoal não fique chateado, pois passamos todas as emoções deste jogo para o computador. Este programa,desenvolvido para o TK 90X, traz tudo isto para dentro de sua casa, possibilitando uma grande disputa em sua família.

O funcionamento do programa é bastante simples, sendo limitado o número de jogadores e, a definição dos gráficos "A" e "B" não trazem dificuldades ao usuário como mostra a tabela 1.

Jogando Boliche no computador

Antes de começar a digitar o programa,o usuário deve definir os gráficos "A" e "B", usando a UDG, que aparecerão nas linhas do programa como ilustra a tabela 2. Para introduzir as figuras nas linhas, onde estas estão programadas, você deve proceder da seguinte maneira: CAPS SHIFT + TECLA 9 (simultaneamente) e depois pressionar a tecla correspondente ao gráfico "A" ou "B".

Tabela 1

Linha 50	-	Gráfico b.
Linha 60	-	Gráfico b.
Linha 80	-	Gráfico b.
Linha 200	-	Gráfico b.
Linha 8020	-	Gráfico a.
Linha 8030	-	Gráfico b.

```
ABCDEFGHIJKLMNOPQRSTU
♦●CDEFGHIJKLMNOPQRSTU
A=♦
```

```
ABCDEFGHIJKLMNOPQRSTU
♦●CDEFGHIJKLMNOPQRSTU
B=●
```

A digitação do programa não oferece maiores dificuldades ao usuário porque, depois de alguns contatos com o equipamento, este dominará por completo o manuseio do teclado.

Ao rodar o programa, a primeira tela a ser apresentada conterá as instru ções do jogo e uma breve explicação, para que o usuário não se sinta perdido diante do equipamento.

Na parte inferior do seu vídeo surgirá uma pergunta, indicando o número de participantes e apresentando como valor mínimo 1 e o máximo 4. Feita a escolha, surgirá a tela de jogo, apresentando na parte direita do vídeo, a pista de jogo com a bola e as garrafas. Do seu lado esquerdo, surgirão quantos jogadores estão participando e seus respectivos pontos. Na parte superior ao canto esquerdo da tela, o número apresentado significa quantas jogadas foram realizadas por participante.

Para dar uma maior animação ao jogo, as cores da tela se alteram quando o jogador passa para o próximo participante.

Para lançar a bola pressione a tecla "1". As teclas "5" e "8" servem para conduzir a bola para direita ou esquerda.

Ao término da partida, a mensagem "FIM DE JOGO" é exibida junto com a pontuação final dos jogadores,perguntando a você se deseja jogar novamente.

```
   1 REM **Boliche**
   3 DIM s(5)
   5 FOR f=0 TO 7: READ a: POKE
USR "a"+f,a: NEXT f
  10 FOR f=0 TO 7: READ a: POKE
USR "b"+f,a: NEXT f
  15 LET x=20: LET k=19
  20 GOSUB 7500
  25 FOR q=1 TO 10
  30 FOR p=1 TO pl
  32 RESTORE 9020+p: READ b,c: B
ORDER b: PAPER c: CLS
  35 GOSUB 8010
  40 FOR t=1 TO 2
  50 FOR k=18 TO 26: PRINT AT x,
k;" B": FOR f=1 TO 3: GOSUB 70:
NEXT f: NEXT k
  60 FOR k=27 TO 19 STEP -1: FOR
 f=1 TO 3: GOSUB 70: NEXT f: PRI
NT AT x,k;"B ": NEXT k: GOTO 50
  70 IF INKEY$<>"1" THEN RETURN
  80 FOR f=x TO 0 STEP -1: PRINT
 AT f,k;"B": IF f>=12 THEN PRINT
 AT f+1,19;"         "
  85 IF f=5 AND k=22 AND t=1 AND
 RND<.5 THEN GOTO 200
  95 PRINT AT f+1,k-1;"   ": IF
f>=12 THEN LET k=k-(INKEY$="5")+
(INKEY$="8"): IF k=18 OR k=28 TH
EN GOTO 110: NEXT f
 100 FOR n=1 TO 4: NEXT n: NEXT
f
 105 GOTO 130
 120 PRINT AT 16,4;"Fora": FOR d
=1 TO 130: NEXT d: PRINT AT 16,4
;"          "
 130 NEXT t
 135 GOSUB 6000
 140 FOR d=1 TO 130: NEXT d
 145 NEXT p
 150 NEXT q
 160 CLS : FOR d=2 TO pl*2 STEP
2: PRINT AT d+5,6; FLASH 1;"Joga
dor"; FLASH 0;d/2; FLASH 1;" Pon
tos= "; FLASH 0;s(d/2): NEXT d:
PRINT AT 1,10; FLASH 1;"Fim de J
ogo"; FLASH 0
 170 INPUT "Deseja Continuar (s/
n) ";a$
 180 IF a$="s" OR a$="S" THEN RU
N
 190 STOP
 200 FOR f=5 TO 0 STEP -1: PRINT
 AT f,19;"     ";"B";"     ";AT f+
1,22;"  ": SOUND .5,f: NEXT f: L
ET s(p)=s(p)+30: GOTO 140
6000 REM ***************
6005 LET s(5)=0
6010 RESTORE 9010
6020 FOR u=1 TO 10: READ a,b: LE
T s(5)=s(5)+(ATTR (a,b)<90): NEX
T u
6025 IF s(5)=10 THEN LET s(5)=15
6030 LET s(p)=s(p)+s(5)
6040 FOR u=2 TO pl*2 STEP 2: PRI
NT AT u+5,10;s(u/2): SOUND .01,p
l*u: NEXT u
6050 RETURN
7500 PRINT AT 3,10; FLASH 1;"INS
TRUCOES"; FLASH 0: PRINT AT 8,3;
"Cada jogador dispoe de 10  joga
das."
7510 PAUSE 50: PRINT AT 10,3;"Se
 o jogador ou um dos jogadores c
onseguir fazer um    STRIKE(derr
ubar todas as garrafas numaso jo
gada) ganhara 30 pontos,  ecada
garrafa derrubada  valera 1ponto
."
7520 PRINT TAB 3;"Pressione ""1"
" para lancar   abola.";TAB 3;"O
 jogador pode controlar    abola
 lancada utilizando as    teclas
 ""5 e 8"""
7530 PAUSE 40: INPUT "Numero de
Jogadores (1-4)";pl: IF pl<1 OR
pl>4 THEN GOTO 7530
7540 CLS : RETURN
8000 REM ***************
8010 RESTORE 9010: PLOT 150,15:
DRAW 0,150: PLOT 151,15: DRAW 0,
150: PLOT 224,15: DRAW 0,150: PL
OT 225,15: DRAW 0,150
8020 FOR f=1 TO 10: READ a,b: PR
INT BRIGHT 1;AT a,b;"A": NEXT f
8030 PRINT ;AT x,k;"B"
8040 PRINT AT 1,0; FLASH 1;q; FL
ASH 0: FOR f=2 TO pl*2 STEP 2: P
RINT AT f+5,0;"Jogador ";f/2;" "
;s(f/2): NEXT f
8050 RETURN
9000 DATA BIN 00011000,BIN 00011
000,BIN 00111100,BIN 00111100,BI
N 00111100,BIN 00011000,BIN 0001
1000,BIN 00000000
9005 DATA BIN 00000000,BIN 00111
100,BIN 01111110,BIN 01111110,BI
N 01111110,BIN 01111110,BIN 0011
1100,BIN 00000000
9010 DATA 2,20,2,22,2,24,2,26,3,
21,3,23,3,25,4,22,4,24,5,23
9021 DATA 2,6
9022 DATA 0,7
9023 DATA 0,6
9024 DATA 2,7
```

List...LList

Simulando os comandos List/LList de seu TK 90X

Este utilitário permite ao usuário selecionar linhas de um determinado programa para impressão. O programa que será listado pode ser de sua autoria ou qualquer outro já existente. "List/LList" será uma boa ferramenta que terá muita utilidade em programas extensos, possibilitando ao usuário determinar por exemplo, o número de linhas que serão impressas (no caso do mesmo estar fazendo algum tipo de verificação).

O utilitário permite ainda, que você escolha o espaço, entre as linhas, para impressão, onde na realidade este programa simula os comandos List/LList de seu TK 90X.

Quais os procedimentos que o usuário deve seguir

O usuário deve fazer um MERGE com o List/LList e o outro já existente ou carregá-lo antes de iniciar a digitação.

Para rodar o programa você deve digitar:

"RUN 9951". Ao rodá-lo, o equipamento pedirá a você para que entre com o número da primeira e da última linha. Em seguida uma nova pergunta onde você deverá responder optando pela impressão em vídeo ou por impressora. Independentemente da opção escolhida, você pode obter uma listagem com espaçejamento simples ou duplo, conforme ilustrações nos exemplos 1 e 2.

No caso do usuário optar pela impressão em vídeo, a listagem será apresentada de forma colorida. Os números que enumeram as linhas aparecerão em vermelho, e o BASIC em azul, mas se você não gostar das cores poderá modificá-las. Para isto bastará apenas modificar as linhas 9985 e 9987, que representam respectivamente, o número da linha e o que está em BASIC.

Um detalhe ao qual o usuário deve ficar atento é ao que se refere ao seguinte tópico:

"o número da primeira linha não pode ser igual ao da segunda. Isto significa que o programa não imprimirá apenas uma linha, e sim no mínimo duas.

Esta ferramenta, que simula dois comandos de uma só vez de seu TK 90X (List/LList), tem grande utilidade na hora de conferir ou corrigir uma listagem qualquer, porque este tipo de tarefa é sempre realizada por etapas, e é neste ponto que se ressalta a vantagem de se ter este utilitário.

```
1     FOR n=1 TO 80
2     PRINT " MICROHOBBY "; "  ";
3     NEXT n
4     GOTO 1
```

.Exemplo 1

```
1     FOR n=1 TO 80

2     PRINT " MICROHOBBY "; "  ";

3     NEXT n

4     GOTO 1
```

Exemplo 2

```
9951 INK 0: PAPER 7: BORDER 7: C
LS
9952 REM LIST/LLIST
9953 INPUT "Primeira linha a ser
 impressa?";Inicio
9954 INPUT "Ultima linha a ser i
mpressa?";Fim
9955 IF Inicio=Fim THEN PRINT "A
 linha inicial e a final devem s
er diferentes": CLS : GOTO 9952
9956 LET prog=23635: LET vars=23
627
9957 DEF FN a(x)=256*PEEK (x+1)+
PEEK x
9958 DEF FN b(x)=PEEK (x+1)+256*
PEEK x
9959 LET prog=FN a(prog)
9960 CLS : PRINT AT 10,10; FLASH
 1;"Aguarde"; FLASH 0
9961 LET vars=FN a(vars)
9962 REM Qual o endereco da linh
a inicial
9963 LET lineadd=prog
9964 IF lineadd=vars THEN CLS :
PRINT "Linha";Inicio;"Nao existe
": STOP
9965 LET lineno=FN b(lineadd)
9966 LET linelen=FN a(lineadd+2)
```

```
9967 IF lineno=Inicio THEN LET s
tlin=lineno: LET stlineadd=linea
dd: LET stlinlen=linelen: GOTO 9
971
9968 LET lineadd=lineadd+4+linel
en
9969 GOTO 9964
9970 REM Qual o endereco da linh
a final
9971 LET lineadd=lineadd+4+linel
en
9972 IF lineadd=vars THEN CLS :
PRINT "Linha";Fim;"Nao existe":
STOP
9973 LET lineno=FN b(lineadd)
9974 LET linelen=FN a(lineadd+2)
9975 IF lineno=Fim THEN LET endl
in=lineno: LET endlineadd=linead
d: LET endlinlen=linelen: GOTO 9
980
9976 LET lineadd=lineadd+4+linel
en
9977 GOTO 9972
9978 PRINT
9979 REM Imprimir Linhas
```

```
9980 INPUT "Imprimir listagem na
 Tela ou na Impressora(t/i)?    "
;q$
9981 CLS
9982 LET dev=2: IF q$="I" OR q$=
"i" THEN LET dev=3
9983 REM Escolha Caracteres para
 Impressao
9984 INPUT "Tipo de Espacejament
o? (1=simples,2=duplo etc.)";sp
9985 PRINT #dev; INK 2;FN b(stli
neadd);TAB 5;
9986 FOR n=stlineadd+4 TO endlin
eadd+2+endlinlen
9987 PRINT #dev; INK 1;CHR$ PEEK
 n AND (PEEK n=13 OR PEEK n>31);
9988 IF PEEK n=14 THEN LET n=n+5
9989 IF PEEK n=13 THEN FOR z=2 T
O sp: PRINT #dev: NEXT z: PRINT
#dev; INK 2;FN b(n+1);TAB 5;: LE
T n=n+4
9990 NEXT n
9991 PRINT #dev
9992 STOP
9999 SAVE "llist" LINE 9951
```

Controle de Estoque no TK 90X

Este aplicativo foi desenvolvido para ser usado por pequenas empresas que possuam um TK 90X e até 100 produtos diferentes em estoque. E fornece um resumo, com dados sobre estoque mínimo necessário, estoque atual e quantidade a ser encomendada, permitindo também a gravação e impressão do arquivo gerado. Para edição do mesmo, cada registro é acessado pelo seu número e não por seu código. O registro pode ser alterado ou deletado. Toda vez que este for alterado grave-o novamente. Quando for iniciá-lo utilize o comando GOTO 1000, nunca o comando RUN. Isso porque as matrizes dos valores, de cada registro, estão dimensionadas no início do programa (não poderia ser de outra forma!). Se ele é reiniciado pelo comando RUN, todas as matrizes, contendo os dados introduzidos, serão redimencionadas, fazendo com que todos os valores sejam zerados.

A lógica empregada neste programa é do tipo modular, ou seja, várias sub-rotinas gerenciadas por GOSUB. As linhas de decisão (Opções do MENU) estão também no início, após a rotina de formatação do MENU (linhas 1000 a I090).

Vale ressaltar um detalhe interessante de seu algorítimo. O número do código do produto (b$) é tratado como STRING e armazenado numa matriz alfabética r$(100,12). Cada linha da matriz armazenará, um código de produto. Sendo assim, cada elemento será um caractere de cada um dos códigos.

Este processo é feito com o comando no formato: LET r$ (a,1 TO12)=b$ que significa: "atribua cada caractere de b$ (de 1 a 12 no máximo) a cada elemento da linha a".

Por exemplo, digamos que você introduziu os códigos: 101010, 202020, 303030.

A matriz ficara na forma:

r$ = [a_{11} =1 a_{12} =0 a_{13} =1 a_{14} =0 a_{15} =1 a_{16}]

r$ =

$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$	$a_{1,6}$	$a_{1,7}$	$a_{1,8}$	$a_{1,9}$	$a_{1,10}$	$a_{1,11}$	$a_{1,12}$
"	"	"	"	"	"	"	"	"	"	"	"
1	0	1	0	1	0	1	0	1	0	1	0
$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$	$a_{2,6}$	$a_{2,7}$	$a_{2,8}$	$a_{2,9}$	$a_{2,10}$	$a_{2,11}$	$a_{2,12}$
2	0	2	0	2	0	2	0	2	0	2	0
$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$	$a_{3,6}$	$a_{3,7}$	$a_{3,8}$	$a_{3,9}$	$a_{3,10}$	$a_{3,11}$	$a_{3,12}$
3	0	3	0	3	0	3	0	3	0	3	0
...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...

Estaremos dessa forma, limitando o número de código a 12 caracteres. É lógico que não é essa a vantagem.

Para cada caractere numérico introduzido na memória, o computador reservará dois bytes. Mas, para cada caractere alfanumérico ele reservará apenas 1. Então, ao tratarmos um número como STRING, economizaremos inúmeros Bytes, pois da forma descrita, o campo código do produto ocupará 12 Bytes no máximo.

Se esse procedimento não fosse tomado ocuparia sempre 24 Bytes!

Ao digitar o programa, para obtenção dos caracteres em português, utilizaremos o recurso de redifinição de caracteres UDG, residente na ROM do TK 90X. Na utilização de todos os caracteres em Português, maísculos e minúsculos devemos acessar UDG0 e logo após UDG1, como linha direta.

Esta operação deve ser feita antes de iniciarmos a digitação. Quando gravado o programa é automaticamente verificado.

```
   1 REM Controle de Estoque
   2 REM ******TK90X*******
  10 DIM r$(100,12): DIM w(100):
 DIM x(100): DIM y$(100,9)
  20 LET a=0: LET i=2
  30 BORDER 0: PAPER 0: INK 4: C
LS
  40 GOSUB 9000
1000 BORDER 0: PAPER 0: INK 4: C
LS
1010 PRINT AT 4,4;"**********MEN
U*********"
1020 PRINT AT 6,4;"TECLA A-ALTER
AR ARQUIVO"
1030 PRINT AT 8,4;"TECLA D-DELET
AR REGISTRO"
1040 PRINT AT 10,4;"TECLA E-ENTR
AR REGISTRO"
1050 PRINT AT 12,4;"TECLA N-NIVE
L DE ESTOQUE"
1060 PRINT AT 14,4;"TECLA C-CARR
EGAR ARQUIVO"
1070 PRINT AT 16,4;"TECLA S-SALV
AR ARQUIVO"
1080 PRINT AT 18,4;"***********
***********"
1090 PRINT AT 21,8;"QUAL SUA OPÇ
ÃO ?": INPUT a$
1100 CLS
1110 IF a$="e" THEN GOTO 3000
1120 IF a$="n" THEN GOTO 4000
1130 IF a$="c" THEN GOTO 5000
1140 IF a$="s" THEN GOTO 6000
1150 IF a$="d" THEN GOTO 7000
1160 IF a$="a" THEN GOTO 2000
1170 GOTO 1090
2000 LET o=1: CLS
2010 PRINT AT 21,0;"ENTRE O CODI
GO A SER ALTERADO": INPUT b
2020 PRINT AT 21,0;"
                             "
2030 LET o=o+1: PRINT AT 0,3;"PR
ODUTO";AT 0,20;"NIVEL": PRINT AT
 1,0;"No.";AT 1,6;"Código";AT 1,
14;"Mín.*Atual*Encargo"
2040 FOR z=1 TO a: IF z=b THEN I
NK 6: PRINT AT 21,0;z;TAB 3;r$(z
,1 TO 12);TAB 16;w(z);TAB 20;x(z
);TAB 23;y$(z)
2050 NEXT z
2060 INPUT g: LET w(b)=w(b)+g
2070 PRINT AT o+1,0;b;TAB 3;r$(b
,1 TO 12);TAB 16;w(b);TAB 20;x(b
);TAB 23;y$(b)
2080 PRINT INK 5;AT 21,0;"Tecla
BREAK-proximo registro"
2090 IF INKEY$="m" THEN GOTO 100
0
2100 IF INKEY$<>" " THEN SOUND .
1,3: PRINT AT 21,0; INK 9;"Tecla
 BREAK-Proximo registro": SOUND
.1,2: GOTO 2080
2110 GOTO 2010
3000 PRINT AT 0,3;"PRODUTO";AT 0
,20;"NIVEL": PRINT AT 1,0;"No.";
AT 1,6;"Código";AT 1,14;"Mín.*At
ual*Encargo"
3010 INPUT b$: SOUND .1,2: LET a
=a+1: LET r$(a,1 TO 12)=b$
3020 INPUT b: SOUND .1,2: LET w(
a)=b: INPUT c: SOUND .1,1: LET x
(a)=c: INPUT d$: SOUND .1,2: LET
 y$(a)=d$
3030 PRINT AT i,0;a;TAB 3;b$;TAB
 14;b;TAB 19;c;TAB 25;d$: PRINT
AT 21,0; INK 4;"Tecla ENTER-Prox
imo registro": SOUND .5,0: PRINT
 INK 4;AT 21,0;"Tecla ENTER-Prox
ima entrada ": SOUND .5,1: INPUT
 a$: IF a$="m" THEN GOTO 1000
3035 IF a$<>"" THEN GOTO 3030
3040 PRINT INK 7;AT 21,0;"
                             ": LET
 i=i+1: GOTO 3010
4000 PRINT AT 0,3;"PRODUTO";AT 0
,20;"NIVEL": PRINT AT 1,0;"No.";
AT 1,6;"Código";AT 1,14;"Mín.*At
ual*Encargo"
4010 FOR z=1 TO a: IF r$(z,1 TO
12)="            " THEN GO
TO 4040
4020 PRINT z;TAB 3;r$(z,1 TO 12)
;TAB 16;w(z);TAB 20;x(z);TAB 24;
y$(z): NEXT z
4040 NEXT z
4050 PRINT INK 5;AT 21,0;"
    TECLA M - MENU        ": I
F INKEY$="m" THEN GOTO 1000
4060 IF INKEY$="z" THEN COPY
4070 GOTO 4050
5000 PRINT AT 0,3;"PRODUTO";AT 0
,20;"NIVEL": PRINT AT 1,0;"No.";
AT 1,6;"Código";AT 1,14;"Mín.*At
ual*Encargo"
5010 FOR z=1 TO a
5020 IF w(z)<x(z) THEN PRINT z;T
AB 3;r$(z,1 TO 12);TAB 16;w(z);T
AB 20;x(z);TAB 23;y$(z): NEXT z
5030 NEXT z
5035 INPUT u$
5040 GOTO 1000
6000 CLS : PRINT AT 21,0;"Nome d
o Arquivo": INPUT a$: SAVE a$:
PRINT AT 21,0;"Término de gravaç
ão.Pressione uma tecla para veri
ficação": INPUT t$: VERIFY a$: P
RINT AT 21,0;"Verificado
              ": INPUT t$: GOTO 1
000
7000 CLS : INPUT "Entre o numero
 do registro a seralterado";p
7010 FOR q=1 TO a: IF q=p THEN G
OTO 7030
7020 NEXT q: PRINT AT 21,0; INK
5;"Codigo ";p;"Não Arquivado
      ": INPUT a$: GOTO 1000
7030 LET r$(q)=r$(a+1): LET w(q)
=w(a+1): LET x(q)=x(a+1): LET y$
(q)=y$(a+1): PRINT AT 21,0; INK
6;"Registro deletado.Pressione E
NTER         ": INPUT a$: GOTO
 1000
9000 PRINT INK 4;TAB 8; FLASH 1;
"CONTROLE DE ESTOQUE"; FLASH 0;T
AB 8;"*******************"
9010 PRINT : PRINT INK 7;TAB 8;
9020 PRINT : PRINT TAB 2;"Toda v
ez que alterar os regis-tros,gra
ve-os novamente(opção S do MENU)
. Os dados são gravados juntamen
te com o programa."
9030 PRINT TAB 2;"Quando usar um
programa com da-dos gravados não
 digite RUN, mas GOTO 1000."
9040 PRINT AT 21,0;"Pressione qu
alquer tecla": IF INKEY$="" THEN
 GOTO 9040
9050 RETURN
9998 SAVE "CTRL.ESTQ.": GOTO 999
8
9999 BORDER 7: PAPER 7: INK 0
```

Investimento Pessoal

Christiano Nasser
e Wilson José Tucci

Hoje em dia, em um País como o Brasil, onde existem altas taxas de inflação, torna-se muito importante uma correta aplicação do nosso dinheiro. Em muitas ocasiões precisamos até de especialistas para analisar os melhores investimentos.

O programa aqui apresentado tem a finalidade de calcular algumas variáveis financeiras, a partir de valores conhecidos. Iremos usar as seguintes variáveis:

- C - Capital no início das contagens dos períodos de capitalização
- CF - Capital no final das contagens dos períodos de capitalização
- P - Número de períodos de capitalização
- J - Taxa de juros por período de capitalização
- I - Taxa de juros compostos em "P" períodos de capitalização
- K - Série uniforme de pagamentos

Vejamos como funciona o programa

"Investimento Pessoal" é dividido em sete blocos que oferecem sete opções de cálculo.

No primeiro bloco o usuário pode calcular o capital final, a partir das variáveis "C", "P" e "J".

Exemplo: "Quanto teremos daqui a 12 meses se aplicarmos, hoje, Cr$ 45 mil a 9,8% ao mês?"

O resultado pode ser obtido usando-se a fórmula: CF=C*(1+J/100)^ P.

O segundo bloco calcula o capital inicial a partir de "CF", de "J" e de "P".

Exemplo: "Quanto devemos pagar hoje para recebermos Cr$ 200 mil daqui a três anos, a juros de 214% ao ano?

Para sabermos o resultado devemos usar a fórmula C=CF/(1+J/100)^ P.

O terceiro bloco calcula o número de períodos a partir de "CF", de "C" e de "J". A fórmula neste caso é a seguinte:

P = (LOG (CF/C)) / (LOG(1+J / 100))

O quarto bloco calcula o capital inicial a partir de "K", de "J" e de "P".

Exemplo: "Uma pessoa quer um rendimento anual de Cr$ 200 mil nos próximos seis anos, sabendo que a taxa de juros é de 234% ao ano. Qual será o valor do investimento?"

Podemos então, utilizar a fórmula: C=K*(((J/100) / (1+J / 100) ^ (-P)))^(-1).

O quinto bloco calcula a taxa de juros por período de capitalização a partir de "P" e de "I". A fórmula é a seguinte:

J = ((1+I / 100) ^ (I/P)-1) * 100

O sexto bloco calcula a série uniforme de pagamentos a partir de "CF",de "J" e de "P". A fórmula usada é:

K = CF * (((1 + J/100)^N-1) / (J/100)) ^ (-1)

O sétimo bloco calcula o número de períodos de capitalização a partir de "K", de "CF" e de "J". Usamos aí, a fórmula:

P = (LOG((CF*J / (100*K))+1)) / LOG (1 + J / 100)

Quando o programa for rodado, o micro lhe perguntará por um bloco. Você deverá escolher entre os sete blocos já citados, o que mais lhe interessa.

Realizados todos os cálculos basta agora o seu bom "faro" para realizar bons investimentos.

```
10  HOME
20  HTAB 12: PRINT "INVES
TIMENTO PESSOAL"
25  VTAB 5
31  PRINT "CALCULAR VARIA
VEIS": PRINT
32  PRINT "1-CAPITAL FINA
L C,P,J"
33  PRINT "2-CAPITAL INIC
IAL CF,J,P"
34  PRINT "3-NUMERO/PERIO
DOS CF,C,J"
35  PRINT "4-CAPITAL INIC
IAL K,J,P"
36  PRINT "5-JUROS/PERIOD
O P,I"
37  PRINT "6-SERIE/PAGAME
NTOS CF,J,P"
38  PRINT "7-NUMERO/PERIO
DO K,CF,J"
39  PRINT : PRINT "FACA S
UA OPCAO (1-7)": GET B
40  ON B GOTO 100.200.300
,400,500,600,700
100  REM  **BLOCO DO CALC
ULO DO CAPITAL FINAL**
110  HOME
115  HTAB 15: PRINT "CAPI
TAL FINAL": PRINT
120  INPUT "ENTRE O VALOR
 DO CAPITAL DE INICIO";C
130  PRINT : INPUT "ENTRE
 O VALOR DO NUMERO DE PER
IODOS";P
140  PRINT : INPUT "ENTRE
 O VALOR DA TAXA DE JUROS
/PERIODO";J
150  REM  **CALCULO DO VA
LOR**
160 CF = C * (1 + J / 100
) ^ P
170  REM  **RESULTADO**
180  HOME : VTAB 5: PRINT
 "O VALOR DO CAPITAL FINA
L E'-->";CF
190  GOTO 1000
200  REM  **BLOCO DO CALC
ULO DO CAPITAL INICIAL**
210  HOME : HTAB 15: PRIN
```

```
T "CAPITAL INICIAL"
220  PRINT : INPUT "ENTRE
 O VALOR DO CAPITAL FINAL
";CF
225  PRINT : INPUT "ENTRE
 O VALOR DA TAXA DE JUROS
";J
230  PRINT : INPUT "ENTRE
 O VALOR DO NUMERO DE PER
IODOS";P
240  HOME
250  REM  **CALCULO DO VA
LOR**
260 C = CF / (1 + J / 100
) ^ P
270  VTAB 5: PRINT "O VAL
OR DO CAPITAL INICIAL E'-
->";C
280  GOTO 1000
300  REM  **BLOCO DO CALC
ULO DO NUMERO DEPERIODOS*
*
310  HOME : HTAB 15: PRIN
T "NUMERO/PERIODOS"":PRIN
T
320  INPUT "ENTRE O VALOR
 DO CAPITAL FINAL";CF: PR
INT
330  INPUT "ENTRE O VALOR
 DO CAPITAL INICIAL";C: P
RINT
340  INPUT "ENTRE O VALOR
 DA TAXA DE JUROS/PERIODO
";J
350  HOME
360  REM  **CALCULO DO VA
LOR**
370 P = ( LOG (CF / C)) /
 ( LOG (1 + J / 100))
380  VTAB 5: PRINT "O VAL
OR DO NUMERO/PERIODOS E'-
->";P
390  GOTO 1000
400  REM  **BLOCO DO CALC
ULO DO CAPITAL INICIAL**
410  HOME : HTAB 15: PRIN
T "CAPITAL INICIAL": PRIN
T
420  INPUT "ENTRE O VALOR
 DA SERIE/PAGAMENTOS";K:
PRINT
430  INPUT "ENTRE O VALOR
 DA TAXA DE JUROS/PERIODO
S";J: PRINT
440  INPUT "ENTRE O VALOR
 DO NUMERO/PERIODOS";P
445  HOME
450  REM  **CALCULO DO VA
LOR**
460 C = K * (((J / 100) /
 (1 + J / 100) ^ ( - P)))
 ^ ( - 1)
470  VTAB 5: PRINT "O VAL
OR DO CAPITAL INICIAL E'-
->";C
480  GOTO 1000
500  REM  **CALCULO DO VA
LOR DA TAXA DE JUROS/PERI
ODO**
510  HOME : HTAB 13: PRIN
T "JUROS/PERIODO": PRINT

520  INPUT "ENTRE O VALOR
 DO NUMERO/PERIODOS";P: P
RINT
530  INPUT "ENTRE A TAXA
DE JUROS COMPOSTOS";I
540  HOME
550  REM  **CALCULO DO VA
LOR**
560 J = ((1 + I / 100) ^
(1 / P) - 1) * 100
570  VTAB 5: PRINT "O VAL
OR DA TAXA DE JUROS E'-->
";J
600  REM  **BLOCO DO CALC
ULO DA SERIE UNIFORME DE
PAGAMENTOS**
610  HOME : HTAB 12: PRIN
T "SERIE/PAGAMENTOS": PRI
NT
620  INPUT "ENTRE O VALOR
 DO CAPITAL FINAL";CF: PR
INT
630  INPUT "ENTRE O VALOR
 DA TAXA DE JUROS";J: PRI
NT
640  INPUT "ENTRE O VALOR
 DO NUMERO DE PERIODOS";P

650  HOME
660  REM  **CALCULO DO VA
LOR**
670 K = CF * (((1 + J / 1
00) ^ P - 1) / (J / 100))
 ^ ( - 1)
680  VTAB 5: PRINT "O VAL
OR DA SERIE E'-->";K
690  GOTO 1000
700  REM  **BLOCO QUE CAL
CULA O NUMERO DEPERIODOS
DE CAPITALIZACAO**

710  HOME : HTAB 14: PRIN
T "PERIODOS": PRINT
720  INPUT "ENTRE O VALOR
 DA SERIE DE PAGAMENTOS";
K: PRINT
730  INPUT "ENTRE O VALOR
 DO CAPITAL FINAL";CF: PR
INT
740  INPUT "ENTRE O VALOR
 DA TAXA DE JUROS";J
750  HOME
760  REM  **CALCULO DO VA
LOR**
770 P = ( LOG ((CF * J /
(100 * K)) + 1)) /  LOG (
1 + J / 100)
780  VTAB 5: PRINT "O NUM
ERO DE PERIODOS  E'-->";P

1000  REM  **FINAL DO PRO
GRAMA**
1010  VTAB 10: PRINT "DES
EJA OUTRO CALCULO(S/N)":
GET B$
1020  IF B$ = "S" THEN  R
UN
1030  HOME : END
```

Gráficos no TK 90X

Victor Mirshawka
Editora Nobel

A elaboração de gráficos por computador é algo que interessa a todos, quer para hobby quer para uso profissional.

Criar uma tela interessante para um jogo, traçar o gráfico de uma função e até um rápido desenho animado podem ser conseguidos por aquele que dominam as técnicas do uso de gráficos no computador.

O TK 90X é um micro computador bastante versátil no que diz respeito a gráficos e utilizá-los é o que ambicionam a maioria de seus usuários.

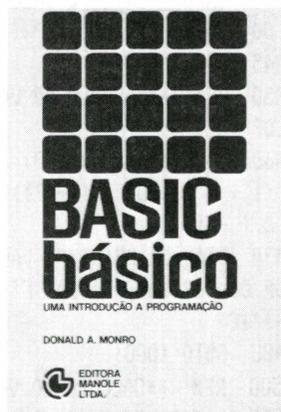
Para introduzí-los a este vasto universo, a editora Nobel está lançando o livro *Gráficos no TK 90X*, de autoria do professor Vitor Mirshawka.

O texto é leve, ideal para um livro introdutório. Os programas são interessantes e de fácil digitação, e bastante detalhados por comentários ao longo de sua listagem.

Os efeitos gráficos conseguidos são bonitos, mas o livro não se prende apenas na estética. Apresenta ao leitor como fazer gráficos de funções, histogramas, distribuições estatísticas, preocupando-se mais com a forma de como obtê-los do que com uma explicação detalhada da teoria envolvida. Mas afinal, é isso que o leitor espera de um livro introdutário: prática acima de tudo.

Gostaríamos contudo de fazer uma ressalva, não ao texto propriamente, mas à sua apresentação: o visual interno é bastante fraco, composição tipográfica pobre (o texto parece uma cópia de uma datilografia feita às pessoas), ilustrações amadorísticas, "listagens" datilografadas... A.A.L.D

BASIC Básico uma introdução à programação

Donald A. Monro
Editora: Manole

Mais um livro introdutório de BASIC. Bem escrito, didático, mas apenas mais um.

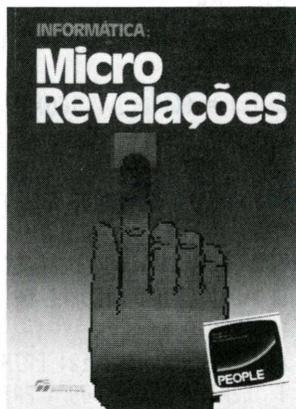
Destinado a principiantes e sem se fixar em nenhuma máquina, o livro aborda passa a passo, como se fosse um curso, todos os principais conceitos de programação e da linguagem BASIC. O texto é leve, didático e é uma boa opção entre os textos introdutórios que pululam no mercado, porém não traz novas abordagens sobre o tema.

A informática é o assunto do momento e suscita uma visualização deque o tema é importante e que vale a pena lançar novos livros sobre o assunto. Certamente vale. Mas deve escolher-se o tema certo, o assunto que o leitor está precisando no momento. Principiantes sempre haverá. Novas abordagens também. Porém, é preciso estar atento. A.A.L.D.

Informática Micro Revelações

Eduardo O.C.Chaves
Editora: Cartgraf/edição conjunta com a People Computação

O primeiro volume da série "Domine o Computador" traz aos usuários de micros pessoais toda uma gama de informações acerca das principais questões que envolvem seu uso.

A principal caraterística do livro é a escolha rigorosa dos diversos textos apresentados, que falam de todos os aspectos introdutórios da informática.

"Micro Revelações" traz para o leitor as implicações da informatização da sociedade e em que setores ela está sendo mais utilizada; o histórico do desenvolvimento da tecnologia, falando sobre o advento do chip, e da formação do vale do Sílicio até o aparecimento dos grandes fabricantes de micros como a Apple Computer.

Na sequência, os autores vão abordando os conceitos básicos da computação (dados alfanuméricos, numéricos, etc.); dos diversos hardware e suas diferentes configurações e capacidades; periféricos, aplicações e extensões de hardware como por exemplo a formação de redes de computadores, assim por diante.

A parte final, os autores dedicaram à introdução das técnicas iniciais de programação, explicando as etapas deste processo, desde a identificação e definição do problema, desenvolvimento da solução logica à documentação.

De todo o conteúdo do livro, (muito bem estruturado) é importante destacar o cuidado com que foi editado.

Desde o texto, muito bem escrito de forma clara e precisa, até a apresentação visual, "Domine o computador" merece um destaque especial pela atenção com que foi criado. Esperamos que os demais volumes da série não percam a qualidade apresentada no primeiro. **A.L.A.**

Projetos de computadores digitais

Glen Langdon Jr
Editora Cartgraf/Scopus

Álvaro A.L. Domingues

Iniciando a Série Tecnologia, com a colaboração da Scopus, a Cartgraf está lançando Projeto de Computadores, *de autoria do Dr. Glen Langdon Jr.*

Projeto de Computadores, *destinado tanto a profissionais que querem se atualizar em novas técnicas ou ter uma boa fonte de consultas, como para estudantes de graduação de um curso de engenharia ou de ciências de computação, é um trabalho oportuno para um país que quer independência tecnológica, pois a independência está no projeto e não na montagem ou cópia de um modelo já pronto.*

O livro integra conceitos de organização de dados em computadores aos métodos práticos do projeto digital propriamente dito. Esta maneira de abordar facilita a compreensão do estudante, mostrando a ele a importância dos conceitos em relação ao projeto global e a cada uma de suas partes, bem como estes conceitos funcionam a nível prático.

O objetivo do livro é ensinar o projeto de um sistema digital não trivial, ou seja, indo além dos circuitos básicos combinacionais e sequenciais.

Um livro neste estilo poderia fácilmente ficar obsoleto, dado o rápido avanço da tecnologia eletrônica no que se refere à microinformática e à criação de novos chips.

Entretanto, mesmo com o advento de novas tecnologias, Projetos de Computadores, *dificilmente, ficará obsoleto. Isso devido à filosofia de ensino empregado: ensinar a pensar e a projetar a nível de unidades funcionais independentes da tecnologia e não como um projeto de flipflops e portas lógicas interligadas.*

Isso é conseguido com o uso do conceito de microoperação: todos os fenômemos internos do computador são estudados como compostos de microoperações que são executadas simultâneamente ou sequencialmente. Este conceito, surgido há aproximadamente 30 anos, sobrevive até hoje, tendo se adequado bem diversas tecnologias.

Esta sequencia de microoperações precisa ser sincronizada por um clock. Basicamente são considerados dois tipos de clock: de uma fase ou de duas fases. Outros tipos de clock podem ser encarados como variações destes dois.

Estes três elementos: fluxo de dados, microoperações e temporização fazem a linha mestra do livro, em cima das quais o autor vai construindo metodologia de trabalho.

O texto é adequado ao nível, sem ser maçante nem prolixo. A leitura flui devido principalmente à linguagem, mas a diagramação leve contribui bastante para que o leitor não se canse, embora trate-se de um assunto complexo.

Ao longo de seis capítulos e quatro apêndices, o autor expõe toda a teoria básica necessária para o desenvolvimento de projetos. Para auxiliar o estudante, cada capítulo é provido de uma série de exercícios que lhe permitem praticar em casa os conceitos visto em sala de aula. Uma referência bibiográfica ajuda aqueles que desejam aprofundar-se no assunto de cada capítulo.

Para ajudar o profissional que usa o livro como fonte de referência, um indice remissivo permite alcançar-se rapidamente qualquer assunto desenvolvido no livro.

Finalizando, um glossário esclarece a terminologia utilizada.

Um livro como o de Langdon Jr. vem preencher uma lacuna importante no mercado editorial de eletrônica, neste momento importante de capacitação tecnologica que vive o país.

O conceito de derivada

Alvaro A.L. Domingues

Apresentamos nesta edição uma abordagem intuitiva da derivada, mostrando de uma forma suave os novos conceitos com os quais o estudante se deparará no primeiro ano de uma faculdade de Ciências Exatas.

Para o estudante

Se você optou por qualquer curso na área de Exatas, a primeira coisa que irá ver na sua frente é um conceito completamente novo para a maioria dos estudantes: as derivadas e integrais.

Embora isto não seja muito difícil, tratase de uma nova forma de interpretar velhos conceitos. E qualquer mudança em nossa maneira de ver encontra resistência e é visto como algo intransponível.

Pretendemos mostrar, neste artigo, o que cada coisa significa e permitir a você uma aproximação menos dolorosa com estas novas idéias.

Aquiles e a tartaruga

Uma das questões filosóficas que gerava muitas discussões entre os gregos antigos era a da possibilidade ou não, do movimento. Um filósofo desta época, Zenão de Heléia, negava enfaticamente o movimento e provava, por meio de um artifício engenhoso, que não havia movimento.

"Suponhamos", dizia ele, "que Aquiles aposte corrida com uma tartaruga. Por ser Aquiles, muitíssimo mais ágil que a tartaruga, é justo que ele dê uma vantagem à ela".(figura 1)

"Ora, para alcançá-la, Aquiles deverá passar pelo ponto A, que está na metade do caminho entre ele e a tartaruga."

"Por outro lado, o corredor deverá ainda passar pelo ponto B, que está na metade do caminho, entre ele e o ponto A. Para atingir o ponto B, deverá passar por C, que está na metade do caminho entre ele e o ponto B. E assim sucessivamente".

"Como vocês podem ver, Aquiles deverá percorrer infinitos pontos e jamais alcançará a tartaruga."

"Portanto, o movimento não existe," concluiu.

Certamente, Zenão de Heléia não negava a evidência de que podíamos nos mover, mas lançou um argumento que era difícil de ser refutado.

Aparentemente, tratava-se de uma brincadeira ou do simples prazer de provocar discussões, mas isto é o embrião do Cálculo Diferencial e Integral.

AQUILES
TARTARUGA
Vantagem
AQUILES
A
TARTARUGA
AQUILES
B
A
TARTARUGA
AQUILES
C
B
TARTARUGA

FIG. 1 - A corrida entre Aquiles e a tartaruga.

Muitos séculos depois...

Ignorando o que disse Zenão de Heléia, imagine que você está dirigindo velozmente um automóvel. Sua velocidade varia ao longo do percurso, chegando a zero, (quando você atinge finalmente seu destino) desmentindo, na prática, os argumentos do filósofo grego.

Podemos medir então, quantos quilômetros você percorreu após um determinado período de tempo.

Suponha que você percorreu 240 quilômetros em três horas, chegando durante este tempo, ao seu destino. Vamos fazer o mesmo que Zenão fez com o trajeto de Aquiles: dividir seu trajeto em diversas partes, só que de forma regular (figura 2).

Coloquemos os marcos de 10 em 10 quilômetros. Mediremos então, o tempo

Figura 2 - *O trajeto do automóvel dividido em trechos regulares*

gasto pelo automóvel para ultrapassar cada um dos marcos, tendo como ponto de análise um gráfico do espaço, em função do tempo.

Podemos ir diminuindo os intervalos para melhorar a precisão do gráfico, até que seja impossível medir o tempo a ser gasto pelo automóvel, quando ele tiver percorrido o trecho considerado (por exemplo, quando o cronômetro não mais puder ser acionado). Poderemos, a partir daí, construir um gráfico como o da figura 3.

Teoricamente, esta medida pode ser diminuída infinitamente, pois entre dois pontos sempre existirá infinitos (era esta a argumentação do filósofo grego). Se pudéssemos atingir o infinito, a menor distância possível seria *infinitamente pequena* e se chamaria *infinitesimal*.

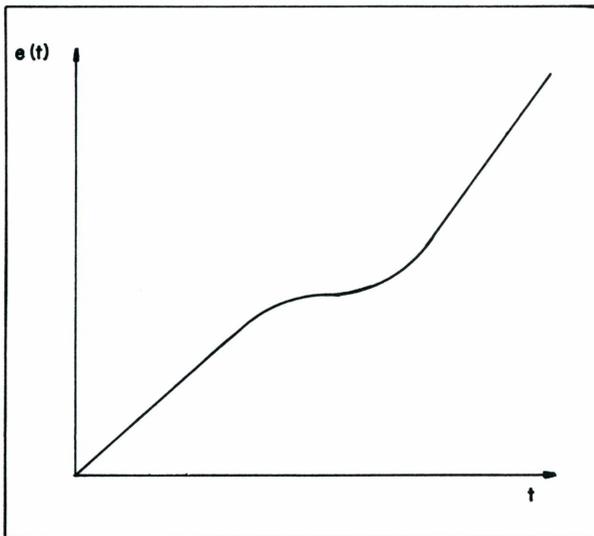



FIG. 3 - Gráfico do espaço percorrido em função do tempo.

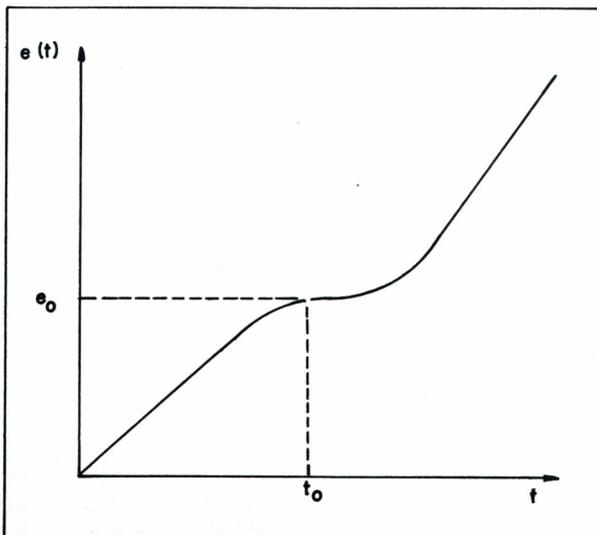



FIG. 4 - Gráfico "furado" em (t_0, e_0).

Os limites

Este primeiro conceito é importantíssimo para o entendimento do conceito de derivada. Mas, antes de prosseguirmos, convém falarmos sobre limites.

Examine novamente o gráfico da figura 3. Suponha que foi tirado *exatamente* um ponto deste gráfico (figura 4). A função ficou "furada", no ponto de coordenadas t_o e_o.

Vamos agora calcular os valores da função,desde o instante zero, aproximando-nos cada vez mais do instante t_0 sem, no entanto, atingí-lo, visto que a função está "furada".

O instante está situado entre dois pontos vizinhos, $t_0 + \delta$ e $t_0 - \delta$ e $t - \delta$ e o valor e_o está entre $e + \epsilon$ $e - \delta$, é proporcional a δ, respectivamente.

Podemos construir em volta do "buraco" uma caixa que o contenha e seja limitada da forma como mostramos na figura 5. Se esta caixa for encolhendo de tal maneira que Δ se aproxime de zero, ϵ se aproximará de zero, t se aproximará de t e_0, finalmente **e** aproximará de e_0. Podemos dizer então, que o limite da função espaço percorrido quando *tende* (aproxima de) t_0 e aproxima-se do valor e_0. Indicamos isso por:

$$\lim_{t \to t_0} e(t) = e_0$$

Pode parecer besteira usar todo este artifício para calcular o valor de um ponto de uma função, mas veja na figura 6 alguns casos em que isto não é tão simples assim.

A derivada

Tomemos o exemplo do automóvel novamente. Você deve estar lembrado do conceito de velocidade média: espaço percorrido, dividido pelo tempo gasto. No nosso caso, a velocidade média vale:

$$v_m = e/t = 240/3 = 80 \text{ km/hora}$$

Podemos calcular a velocidade média entre dois pontos A e B, cujos tempos são tomados entre t_A e t_B. A velocidade média entre eles é:

$$v_{AB} = \frac{e_B - e_A}{t_B - t_A}$$

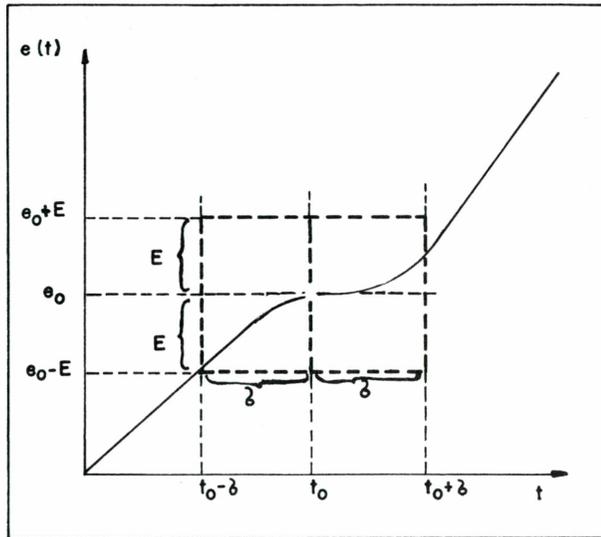



FIG. 5 - Uma "caixa" dentro do gráfico e(t).

FIG. 6 - Onde é difícil calcular o limite.

Isso equivale ao cálculo da inclinação da reta que une os dois pontos A e B (figura 7).

Sabemos também que existe uma velocidade instantânea, (aquela que é medida pelo velocímetro do carro) por exemplo t_0 (figura 8). Podemos ir escolhendo pontos A e B cada vez mais próximos, até que A e B estejam tão próximos que A seja praticamente igual a B. Podemos definir então, velocidade instantanêa como:

$$v(t_0) = \lim_{t_B - t_A} \frac{e(B) - e(A)}{e(B) - t(A)}$$

Generalizando:

$$v(t) = \lim_{\Delta t \to 0} \frac{\Delta e}{\Delta t}$$

Δ significa *diferença*

Podemos então calcular em cada instante, a velocidade do automóvel obtendo o gráfico da figura 9. A velocidade instantânea será o valor da inclinação da reta tangente à curva, no instante considerado.

Você acabou de obter a derivada da função e (t)!

A taxa de variação

Toda esta conceituação foi estabelecida para entendermos certos fenômenos físicos, onde não interessa a função propriamente dita, mas a sua *taxa de variação*.

Por exemplo, a velocidade é a medida da variação do espaço percorrido em função do tempo. Por outro lado, a aceleração é a variação da velocidade em função do tempo.

Alguns fatos merecem uma consideração especial. Em primeiro lugar a sua definição formal:

$$f'(x) = \lim_{\Delta x \to 0} \frac{\Delta f(x)}{\Delta x}$$

$f'(x)$ = derivada da função $f(x)$

Geralmente, os valores envolvidos nesta expressão, quando o limite é atingido, formam uma divisão de zero por zero, mas o valor do limite é diferente de zero. Na derivada é que se verifica a utilidade da teoria dos limites, visto que sempre trabalha-

remos com condições que,na álgebra normal, seriam impossíveis de serem resolvidos.

A importância da derivada e de sua função inversa, a integral. Ela é sentida quando trabalhamos com fenômenos físicos contínuos no tempo como é o caso dos movimentos dos corpos, onde devemos conhecer a sua taxa de variação. Um dos criadores da teoria das derivadas foi Newton quando estudava as leis do movimento, percorrendo um caminho semelhante ao nosso, ou seja, analisando o que acontecia com determinadas funções quando o intervalo entre duas medidas sucessivas era muito pequeno. De quebra, desmontou o argumento de Zenão de Helena.

Para o professor

O programa que apresentamos nesta edição é do tipo mais simples que pode ser desenvolvido em educação: o programa *demonstrativo.*

De um modo geral, podemos dividir os programas educacionais em três tipos básicos:

a) demonstrativos;
b) iterativos de pergunta e resposta;
c) iterativos completos.

Um programa demonstrativo apenas mostra ao estudante um determinado assunto, sem que ele tenha participação ativa, ou restrita no mesmo.

Um programa interativo de perguntas e respostas permite uma participação maior do estudante, visto que ele deve responder a determinadas perguntas apresentadas pelo micro e corrigidas posteriormente, fornecendo um "feed-back" imediato ao aluno.

Programas iterativos completos simulam determinada situação hipotéticas que o estudante deve solucionar. Um exemplo disso é os simulador de vôo que permite um treinamento, em terra, dos pilotos.

A estrutura dos programas demonstrativos são muito simples, porque a única preocupação do usuário deve ser escolher os textos e figuras que deverão ser apresentados; escolher o tempo que estes devem ficar expostos no vídeo e estabelecer uma sequência. Ao contrário dos programas iterativos, a estrutura é linear e a única preocupação é com a disposição na tela.

O programa que apresentamos foi desenvolvido para o TK-2000, mas poderá ser adaptado a outros computadores sem muitos problemas.

Observação: Iteratico = processo de resolução de equação mediante sequência de operações em que o objeto de cada uma é o resultado da que a precede.

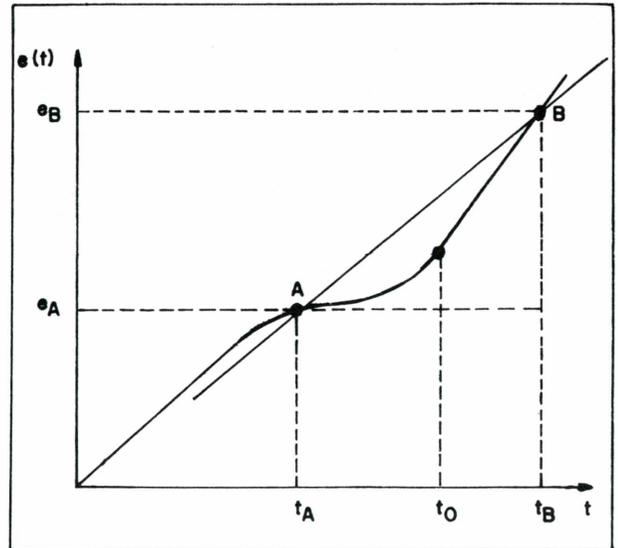



FIG. 7- A velocidade média entre A e B.

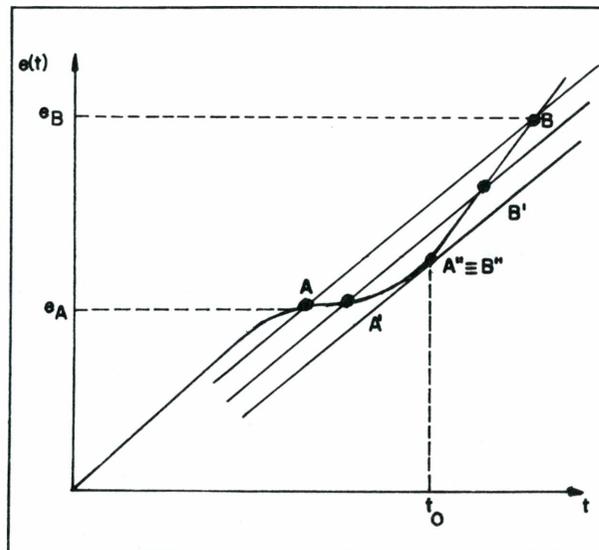



FIG.8 - Calculando a velocidade em um ponto.

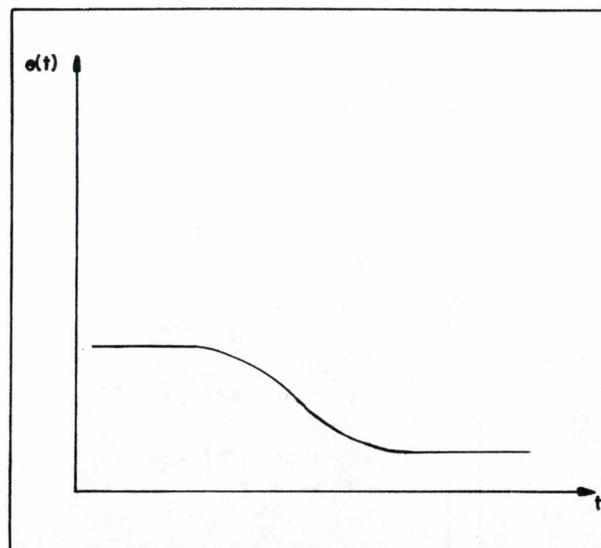



FIG. 9 - A velocidade do automóvel.

```
10  REM   PROGRAMA PARA D
EMONSTRACAO
20  HOME
30  PRINT " ESTE PROGRAMA
 E' DO TIPO DEMONSTRATIVO.
"
40  PRINT " ELE IRA APRES
ENTAR UM ASSUNTO,"
50  PRINT " TOPICO POR TO
PICO, COMO NUM LIVRO"
60  GOSUB 9000
70  PRINT " O ASSUNTO ESC
OLHIDO E':"
75  FOR I = 1 TO 400: NEX
T
80  HTAB (10): VTAB (10):
 PRINT " A DERIVADA"
85  FOR I = 1 TO 500: NEX
T
90  GOSUB 9000
95  HTAB (1): VTAB (10)
100  PRINT " O ASSUNTO EM
 GERAL E'VISTO NO PRIMEIRO
 SEMESTRE DE UM CURSO DE
CALCULO  DIFE- RENCIAL E I
NTEGRAL, DE UM CURSO UNIVE
RSITARIO DA AREA DE EXATAS
"
110  GOSUB 9000
120  PRINT " IMAGINE UMA
FUNCAO,": HTAB (10): VTAB
(10): PRINT "Y=X^2": PRINT
 : PRINT "CUJO GRAFICO TRA
CAREMOS": GOSUB 9000
130  GOSUB 8000
150  PRINT "VAMOS ESCOLHE
R UM VALOR QUALQUER DE X"
160  FOR I = 1 TO 400: NE
XT : PRINT "POR EXEMPLO: X
1"
165  FOR I = 1 TO 2000: N
EXT : HOME
170  HTAB (30): VTAB (20)
: PRINT "X1"
180  HPLOT 210,150 TO 210
,101
190  HPLOT 210,101 TO 140
,101
195  VTAB (13): HTAB (18)
: PRINT "Y1"
200  GOSUB 9000
```

```
210  HTAB (1): VTAB (21):
 PRINT " VAMOS AGORA ESCOL
HER DOIS PONTOS, UM COM X
MAIOR QUE X1 E OUTRO MENOR
"
220  FOR I = 1 TO 2000: N
EXT
230  GOSUB 8000
235 Z = 1
240  HTAB (26): VTAB (20)
: PRINT "X0"
250  HTAB (36): VTAB (20)
: PRINT "X2"
270  HPLOT 180,150 TO 180
,135
280  HPLOT 180,135 TO 140
,135
290  HPLOT 248,150 TO 248
,35
300  HPLOT 248,35 TO 140,
35
310  VTAB (17): HTAB (18)
: PRINT "Y0"
320  VTAB (5): HTAB (18):
 PRINT "Y2"
325  IF Z = 0 THEN  GOTO
350
330  HTAB (1): VTAB (21):
 PRINT "VAMOS TRACAR UMA R
ETA POR ESTES  PONTOS"
340  FOR I = 1 TO 2000: N
EXT : HCOLOR  = 0:Z = 0: G
OTO 240
350  HCOLOR  = 7
360  HPLOT 165,150 TO 250
,30
380  FOR I = 1 TO 2000: N
EXT
390  HTAB (1): VTAB (21):
 PRINT " VAMOS AGORA APROX
IMAR ESTES PONTOS UM DO OU
TRO"
400  FOR I = 1 TO 1000: N
EXT
435 Z = 1
440  HTAB (26): VTAB (20)
: PRINT "  "
450  HTAB (36): VTAB (20)
: PRINT "  "
460  VTAB (17): HTAB (18)
: PRINT "  "
```

```
470  VTAB (5): HTAB (18):
 PRINT "  "
480  FOR I = 1 TO 2000: N
EXT
540  HTAB (28): VTAB (20)
: PRINT "X0"
550  HTAB (32): VTAB (20)
: PRINT "X2"
560  VTAB (16): HTAB (18)
: PRINT "Y0"
570  VTAB (11): HTAB (18)
: PRINT "Y1"
575  HCOLOR  = 0
580  HPLOT 165,150 TO 250
,30
585  HCOLOR  = 7
600 Z = 1
670  HPLOT 195,150 TO 195
,120
680  HPLOT 195,120 TO 140
,120
690  HPLOT 220,150 TO 220
,85
700  HPLOT 220,85 TO 140,
85
710  IF Z = 0 THEN  GOTO
800
720  HTAB (1): VTAB (21):
 PRINT "VAMOS TRACAR UMA R
ETA QUE UNE ESTES PON-TOS
      "
730  FOR I = 1 TO 2000: N
EXT :Z = 0: HCOLOR  = 0: G
OTO 670
800  HCOLOR  = 7
810  HPLOT 170,150 TO 250
,45
820  FOR I = 1 TO 2000: N
EXT
830  TEXT
835  HTAB (1): VTAB (10):
 PRINT " VOCE DEVE TER NOT
ADO QUE :"
840  FOR I = 1 TO 2000: N
EXT
850  HOME : HTAB (1): VTA
B (10): PRINT "X2 APROXIMA
-SE CADA VEZ MAIS DE X1"
860  PRINT : PRINT "Y2 AP
ROXIMA-SE DE Y0"
865  PRINT
```

```
870  HTAB (10): PRINT "(X
2-X0)->0"
875  PRINT
880  HTAB (10): PRINT "(Y
2-Y0)->0"
890  FOR I = 1 TO 2000: N
EXT : HOME
900  HTAB (1): VTAB (10):
 PRINT " POREM, A RAZAO:":
 PRINT
910  PRINT "  (Y2-Y0)/(X2
-X0) "
915  PRINT
920  PRINT " E' DIFERENTE
 DE ZERO "
930  FOR I = 1 TO 2000: N
EXT : HOME
940  HTAB (1): VTAB (10):
 PRINT "ESTA RAZAO  QUANDO
 X2 E' DIFERENTE DE X0"
950  PRINT "E, CONSEQUENT
EMENTE,"
960  PRINT : PRINT " Y2 E
' DIFERENTE DE Y0"
970  PRINT : PRINT " E' A
 MEDIDA DA INCLINACAO DA R
ETA DEFI-NIDA PELOS PONTOS
 (X2,Y2) E (X0,Y0)"
980  FOR I = 1 TO 2000: N
EXT : HOME
990  HTAB (1): VTAB (10):
 PRINT " NO NOSSO CASO, A
RETA E' SECANTE  'A CURVA
X^2": PRINT
1000  PRINT " O QUE ACONT
ECERIA COM A SECANTE SE (X
2-X0) TENDESSE A ZERO ?"
1010  GOSUB 9000
1020  GOSUB 8000
1030  FOR I = 1 TO 2000:
NEXT : HCOLOR  = 7
1035 Z = 1
1040  HTAB (28): VTAB (20
): PRINT "X2=X0"
1050  HPLOT 210,150 TO 21
0,101
1060  HPLOT 210,101 TO 14
0,101
1070  VTAB (13): HTAB (14
): PRINT "Y2=Y1"
1080  IF Z = 0 THEN  GOTO
 1200
1090  FOR I = 1 TO 1000:
NEXT : HCOLOR  = 0:Z = 0:
GOTO 1050
1200  HCOLOR  = 7
1210  HPLOT 175,150 TO 26
5,30
1220  GOSUB 9000
1225  TEXT
1230  HTAB (1): VTAB (10)
: PRINT " A RETA TORNOU-SE
 TANGENTE A CURVA"
1240  FOR I = 1 TO 2000:
NEXT : HOME :
1250  PRINT "PODEMOS DEFI
NIR DERIVADA NUM PONTO COM
O:"
1255  PRINT : PRINT " A I
NCLINACAO DA RETA TANGENTE
 'A CURVA NESTE PONTO"
1260  PRINT : PRINT " EXI
STEM OUTRAS DEFINICOES DE
DERIVADA"
1270  PRINT : PRINT " CON
SULTE BONS LIVROS E CALCUL
O DIFERENCIAL E INTEGRAL P
ARA APOFUNDAR-SE NO ASSUNT
O"
1280  FOR I = 1 TO 3000:
NEXT : HOME
1300  HTAB (1): VTAB (10)
: PRINT " BOA SORTE EM SEU
S ESTUDOS"
7999  END
8000  REM  GRAFICO DA FUN
CAO X^2
8010  HGR : HCOLOR  = 7
8020  HPLOT 140,1 TO 140,
150: HPLOT 1,150 TO 278,15
0
8030  FOR X = 0 TO 11 STE
P .1
8040  LET Y =  INT (X ^ 2
 + 0.5)
8050  HPLOT X * 10 + 140,
150 - Y
8055  HPLOT 140 - X * 10,
150 - Y
8060  NEXT X
8999  RETURN
9000  HTAB (1): VTAB (21)
: PRINT "DIGITE QUALQUER T
ECLA": GET A$: HOME : RETU
RN
```

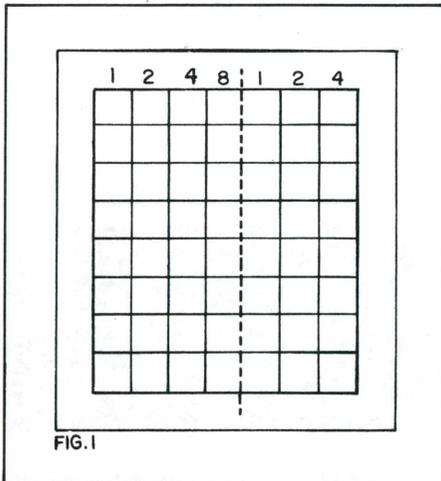
Assembly 6502
Aula VII

Gustavo Egidio de Almeida

Nesta edição, apresentamos um programa muito especial. Tão especial que esta aula será dedicada eclusivamente a ele.

O programa em questão aborda toda a técnica gráfica e em cores que pode ser usado na tela de seu TK 2000.

Com ele você aprenderá todos os macetes e estará dando um grande passo no aprendizado da linguagem Assembly do 6502.

FIG. I

O Byte é representado no vídeo contendo 8 linhas por 07 colunas.

Na alta resolução, cada linha impressa na tela é denominada Microlinha.

Portanto um byte apresenta 8 microlinhas.

No modo baixa resolução a tela é dividida em 40 colunas por 24 linhas.

Tendo um byte situado nessa tela corresponderá a uma posição desta tela, numa respectiva coluna e linha.

No modo alta resolução, o número de colunas continua o mesmo, porém número de linhas, aumenta sensivelmente.

No modo alta resolução, dividimos cada linha do modo baixa resolução por 8 microlinhas, obtendo assim o total de 24 × 8 microlinhas (192 microlinhas) na tela do TK-2000.

Assim, para representar um byte no vídeo, devemos construí-lo de modo a ter 8 linhas por 7 colunas.

A área livre para construção de imagens na tela do televisor pode estar na página um de vídeo (a mais usada) nos endereços $ 2000 a $ 3FFF.

Vamos construir um byte.

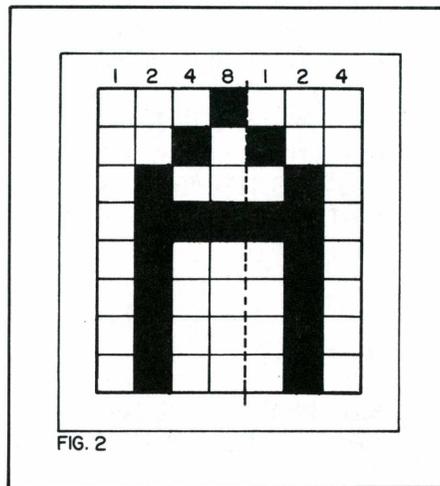
Para construção de bytes que vão formar uma tela, utilizamos folhas de papel quadriculado.

Tomamos agora 8 linhas (8 quadrilinhas) por 7 colunas e obtemos então um byte.

Traçamos agora uma linha divisoria (fig.01), deixando 4 colunas na esquerda e 3 colunas na direita do byte.

Fazendo essa divisão, podemos representar agora cada linha do byte através de um determinado número hexadecimal que será codificado pelo computador.

Vamos representar uma letra qualquer:

FIG. 2

Como vemos, o byte é numerado em sua primeira metade de 1 à 8 e em sua segunda metade de 1 à 4.

Para codificar as linhas, contamos quais os bytes que estão setados (marcados por cruzinhas).

Para formar o byte hexadecimal, pegamos, primeiramente o número formado pelo lado direito do byte e depois o número formado à esquerda deste.

Assim, na primeira linha, temos o valor 08. Na segunda linha, 14. Na terceira, 22. Na quarta, 3E. Na quinta, sexta, sétima e oitava linhas, o valor 22.

Veja que, para formar a quarta linha da caixa teremos que agir da seguinte forma:

Somamos todos os bytes setados da metade direita do byte, obtendo assim o número 2+1=3;

Em seguida somamos os bytes da outra metade do byte, ou seja 8 + 4 + 2 = E (em hexadecimal);

Formamos assim, o valor 3E que representa a quarta linha do byte formado pela letra A.

Vemos então que, para representarmos graficamente um byte na tela, pre cisamos de oito códigos hexadecimais. Porém, como fazemos para obtêlos num determina ponto do vídeo?

Para conseguir isto, devemos recorrer ao "mapa da tela", que está impresso na página 47 do manual técnico do TK-2000.

Observando este mapa, temos os endereços em hexadecimal (coluna da esquerda) de cada início de linha e de microlinha que pode ser observado no quadro à direita em destaque.

Assim, vemos que o primeiro endereço disponível para colocarmos nossa primeira linha de byte, é o valor $2000 da memória.

Vamos então, com o auxílio de um programa, imprimir na tela a Letra **A**.

Programa Assembly - Listagem 1 (desenho letra A)

```
0 800 - 20 32 F8      JSR $ F832
0 803 - AD 54 C0      LDA $ C054
0 806 - AD 50 C0      LDA $ C050
0 809 - A9 08         LDA    $08
0 808 - 8D 00 20      STA $ 2000
0 80E - A9 14         LDA    $14
0 810 - 8D 00 24      STA $ 2400
0 813 - A9 22         LDA    $22
0 815 - 8D 00 28      STA $ 2000
0 818 - A9 3E         LDA    $3E
0 81A - 8D 00 2C      STA $ 2C00
0 81D - A9 22         LDA    $22
0 81F - 8D 00 30      STA $ 3000
0 822 - A9 22         LDA    $22
0 824 - 8D 00 34      STA $ 3400
0 827 - A9 22         LDA    $22
0 829 - 8D 00 38      STA $ 3800
0 82C - A9 22         LDA    $22
0 82E - 8D 00 3C      STA $ 3C00
0 831 - 4C 31 08      JMP $ 0831
```

Vamos analisá-lo linha por linha:
- É chamada a sub-rotina que provoca um HOME na tela.
- Seleciona o vídeo colorido;
- "Seta" a primeira página de vídeo;

- Começa a impressão da letra **A**.

O valor # 08 (que representa a primeira linha do byte), é "jogador" no endereço.

O valor # 14 é armazenado no endereço $ 2000 que representa a segunda linha do byte.

```
# 22 - $ 2800
# 3E - $ 2C00
# 22 - $ 3000
# 22 - $ 3400
# 22 - $ 3800
# 22 - $ 3C00
```

Este programa foi desenvolvido com o intuito de ser mais explicativo possível, não levando em conta o seu tamanho e a economia de bytes. É lógico que pode ter seu tamanho reduzido com o uso de instruções indexadas.

Como vemos, o funcionamento do programa é bem simples não devendo haver portanto, maiores dificuldades no seu aprendizado.

Vamos passar agora um programa todo especial de nossa aula, que utiliza todos os recursos gráficos da tela, inclusive as quatro cores disponíveis: branco, azul, verde e vermelho.

A primeira coisa que devemos fazer quando desejamos imprimir um desenho qualquer na tela, é se utilizar de uma folha de papel quadriculado, para codificar a figura em bytes e seus respectivos bites.

Com o programa que apresentaremos a seguir, será possível codificar a figura ou

FIG. 3

tranpô-la no vídeo de seu computador.

$ 7000 - Apresentaremos agora os caracteres a serem impressos. Serão dados os códigos de cada uma das 32 microlinhas, divididas em conjuntos de 8 bytes cada.

O programa contém uma rotina de impressão inversa, que vai do oitavo para o primeiro byte, do sentido direito para esquerda.

```
$ 7000
- 00  01  60  3C  70  0E  07  00
- 00  02  10  49  08  06  06  00
- 00  04  09  08  05  67  0E  00
- 00  04  09  78  04  44  52  00
- 00  04  08  48  04  44  22  00
- 00  04  09  08  04  44  22  00
- 00  02  11  09  08  44  22  00
- 00  01  63  1C  71  6E  27  00
- 15  00  00  00  00  00  00  2A
- 15  0E  71  71  71  63  47  2A
- 15  04  22  22  22  13  47  2A
- 14  04  24  24  24  09  02  0A
- 14  01  47  67  64  09  7E  0A
- 14  01  02  22  24  09  7E  0A
- 14  01  04  14  14  09  02  0A
- 14  01  02  22  22  13  47  0A
- 14  03  41  71  71  63  47  0A
- 14  00  00  00  00  00  00  0A
- 14  00  00  00  00  00  00  0A
- 14  00  00  0A  55  28  00  0A
- 14  00  05  2A  55  2A  50  0A
- 14  00  05  2A  55  2A  50  0D
- 14  00  55  2F  FF  7E  55  0A
- 14  00  55  71  2A  47  55  0A
- 15  2A  55  55  AA  55  55  2A
- 15  2A  55  55  AA  55  55  2A
- 15  2A  55  55  AA  55  55  2A
- 00  00  55  71  2A  43  55  00
- 00  00  55  2F  FF  7E  55  00
- 00  00  05  2A  55  2A  50  00
- 00  00  05  2A  55  2A  50  00
- 00  00  00  0A  55  28  00  00
- 00  00  00  00  00  00  00  00
- 00  00  00  00  00  00  00  00
```

Rotina Principal

```
- 0 800 - 20 32 F8     JSR $ F832
- 0 803 - AD 54 C0     LDA $ C054
- 0 806 - AD 50 C0     LDA $ C050
- 0 809 - A2 00        LDX # $ 00
- 0 808 - A0 00        LDY # $ 00
- 0 80D - 84 30        STY # $ 30
- 0 80F - A4 30        LDY # $ 30
- 0 811 - B9 00 80     LDA $ 8000,Y
- 0 814 - 8D 0D 09     STA $ 090D
- 0 817 - B9 00 90     LDA $ 9000,Y
- 0 81A - 8D 0C 09     STA $ 090C
- 0 810 - C8           INY
- 0 8LE - 84 30        STY $30
- 0 820 - 20 00 09     JSR $ 0900
- 0 823 - 4C 0F 08     JMP $ 080F
```

Sub-rotina de Impressão na Tela

```
- 0 900 - EA           NOP
- 0 901 - EA           NOP
- 0 902 - EA           NOP
- 0 903 - EA           NOP
- 0 904 - EA           NOP
- 0 905 - EA           NOP
- 0 906 - A0 08        LDY # $08
- 0 908 - BD 00 70     LDA $ 7000,X
- 0 90B - 99 00 00     STA $ 00 00
- 0 90E - E8           INX
- 0 90F - 88           DEY
- 0 910 - D0 F6        BNE $ 0908
- 0 912 - 60           RTS
```

$ 7000 - **Caracteres**

```
- 8000 -  20 24 28    9000 -  00 00
          2C 30 34            00 00
          38                  00 00
          3C                  00 00
          20 24 28            80 80
          2C 30 34            80 80
          38                  80 80
          3C                  80 80
          21 25               00 00
          29 2D               00 00
          31 35               00 00
          39 3D 21            00 00
          25 29               80 80
          20 31 35            80 80
          39 3D               80 80
                              80 80
```

As tabelas contém os caracteres a serem impressos na tela, e os endereços da página de vídeo.

Os caracteres a serem impressos na tela estão a partir do endereço $ 7000. Com os conteúdos dos endereços $ 8000 e $ 9000 agrupados dois a dois, formaremos todos os endereços do vídeo.

Exemplo: $ 8000 - 20 $ 9000 - 00

Formamos o endereço $ 2000.

Digite o programa, rode-o e tente compreender linha por linha o seu funcionamento.

Como este programa é o mais complicado de nosso curso de Assembly, entendemos que você poderá ter alguma dificuldade em compreendê-lo.

Não se importe porém, com o emprego de cores nas figuras, esta técnica será explicada no próximo número, com mais detalhes.

Rode o programa e observe o seu belo efeito.

"Um, Dois, Três...Muitos"

Renato da Silva Oliveira

A partir desta edição passaremos a alternar, mensalmente, a publicação de um Quebra-Cabeça com a de uma Solução. Neste número, estaremos publicando a solução do Quebra-Cabeça "Um, Dois, Três... Muitos". Na próxima edição publicaremos outro Quebra-Cabeça.

Todos os demais Quebra-Cabeças propostos a partir deste mês serão respondidos pelo próprio Nabor ou pela Dinorá ou por algum de nossos amigos.

A seguir, estão listados os dois programas criados pela Dinorá sob os efeitos misteriosos do suco de morangos silvestres hindus do Ramarujan. Para compreender melhor como os programas funcionam, acompanhe seus fluxogramas. A idéia fundamental desses programas é fazer com que, sempre que o valor calculado pelo micro for maior ou igual a dez (10), uma variável contadora seja incrementada uma vez e o valor seja dividido por dez (10). Com isso a variável contadora indicará quantas vêzes o valor final foi dividido por dez, e o valor final será sempre menor que dez. Podemos então escrever o resultado como sendo o valor final, vezes dez, elevado ao conteúdo da variável contadora. Nos programas da Dinorá, o valor final fica na variável [B] e a variável contadora é [A]. O resultado é

$$B \times 10^{A}$$

Fluxograma 1 - *Programa Fatorial;*

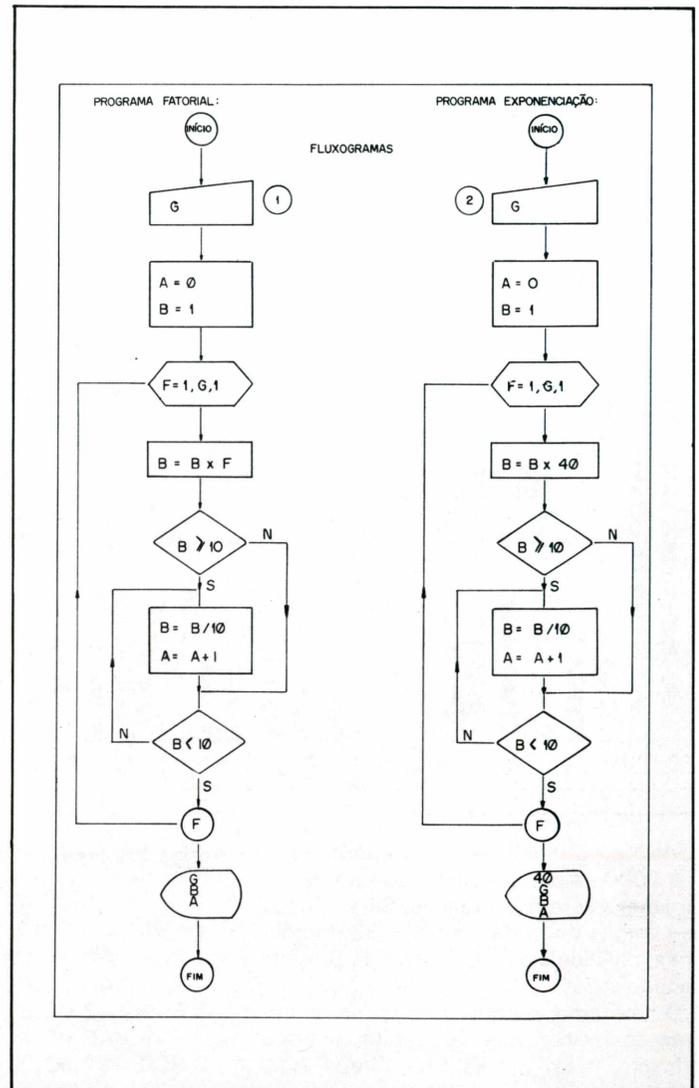
```
FATORIAL

  10 INPUT G
  20 LET A=0
  30 LET B=1
  40 FOR F=1 TO G
  50 LET B=B*F
  60 IF B<10 THEN GOTO 90
  70 LET B=B/10
  80 LET A=A+1
  90 IF B>=10 THEN GOTO 70
 100 NEXT F
 110 PRINT "O FATORIAL DE ";G;"
E ";B;TAB 32;"VEZES 10 ELEVADO
A POTENCIA ";A;"."
```

Fluxograma 2 - *Programa exponenciação.*

```
EXPONENCIACAO

  10 INPUT G
  20 LET A=0
  30 LET B=1
  40 FOR F=1 TO G
  50 LET B=B*40
  60 IF B<10 THEN GOTO 90
  70 LET B=B/10
  80 LET A=A+1
  90 IF B>=10 THEN GOTO 70
 100 NEXT F
 110 PRINT "40 ELEVADO A ";G;" E
 ";B;TAB 32;"VEZES 10 ELEVADO A
POTENCIA ";A;"."
```

Garantia integral

MICROSOFT
MICROSOFT

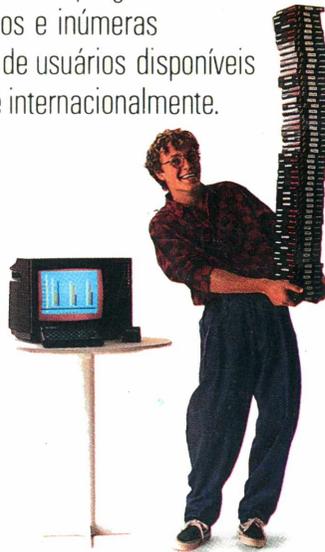
A Microsoft tem 120 programas em fitas e disquetes à sua disposição. São sistemas aplicativos para acompanhar e agilizar os negócios de sua empresa. E também jogos eletrônicos para você e sua família se divertirem muito. Todos especiais para TK-83, TK-85, TK-2000, Apple II e compatíveis. E todos com a mesma qualidade dos 100.000 programas já vendidos em todo o Brasil. Procure o revendedor Microsoft mais próximo (se não encontrar os programas Microsoft escreva para a Caixa Postal 54221 - CEP 01000- S. Paulo-SP). Você encontrará os melhores programas da sua vida.

MICROSOFT®
Sempre o melhor programa.

A Microdigital lança no Brasil o micro pessoal de maior sucesso no mundo.

A partir de agora a história dos micros pessoais vai ser contada em duas partes: antes e depois do TK 90X.

O TK 90X é, simplesmente, o único micro pessoal lançado no Brasil que merece a classificação de "software machine": um caso raro de micro que pela sua facilidade de uso, grandes recursos e preço acessível recebeu a atenção dos criadores de programas e periféricos em todo o mundo.

Para você ter uma idéia, existem mais de 2 mil programas, 70 livros, 30 periféricos e inúmeras revistas de usuários disponíveis para ele internacionalmente.

E aqui o TK 90X já sai com mais de 100 programas, enquanto outros estão em fase final de desenvolvimento para lhe dar mais opções para trabalhar, aprender ou se divertir que com qualquer outro micro.

O TK 90X tem duas versões de memória (de 16 ou 48 K), imagem de alta resolução gráfica com 8 cores, carregamento rápido de programas (controlável pelo próprio monitor), som pela TV, letras maiúsculas e minúsculas e ainda uma exclusividade: acentuação em português.

Faça o seu programa: peça já uma demonstração do novo TK 90X.

Preço de lançamento*
16 K - Cr$ 1.899.850 • 48 K - Cr$ 2.199.850

Chegou o micro cheio de programas.

FOX

Filiada à ABICOMP

Validade: 30/11/85
Valor ☐ Assinatura Inicial: Cr$ 78.000,00
☐ Renovação : Cr$ 65.000,00

Av. ANGÉLICA, 2318 — 14º Andar
PABX — 255-0366 — Cx. Postal: 54096
CEP: 01228 — SÃO PAULO — SP

ATENÇÃO:
EM CASO DE RENOVAÇÃO DE ASSINATURA COLE A ETIQUETA DE ENDEREÇAMENTO ATUAL NO ESPAÇO RESERVADO AO ENDEREÇO, VIA MICROHOBBY.

ASSINANTE: ....
....
....
....

ENDEREÇO: ....
....
....
....

VALIDADE: 30/11/85
VALOR
☐ ASSINATURA INICIAL — Cr$ 78.000,00
☐ RENOVAÇÃO — Cr$ 65.000,00

CREDITAR:
BRADESCO — AG. CONSOLAÇÃO —
C/C Nº 73.966-9

VÁLIDO SE AUTENTICADO MECANICAMENTE PELO BANCO

VIA ASSINANTE

ASSINANTE: ....
....
....
....

ENDEREÇO: ....
....
....
....

VALIDADE: 30/11/85
VALOR
☐ ASSINATURA INICIAL — Cr$ 78.000,00
☐ RENOVAÇÃO — Cr$ 65.000,00

CREDITAR:
BRADESCO — AG. CONSOLAÇÃO —
C/C Nº 73.966-9

VÁLIDO SE AUTENTICADO MECANICAMENTE PELO BANCO

VIA BANCO

AUTORIZO PELO PRESENTE MINHA:
☐ ASSINATURA INICIAL — Cr$ 78.000,00
☐ RENOVAÇÃO — Cr$ 65.000,00
DA REVISTA MICROHOBBY POR 12 EDIÇÕES.
ESTES PREÇOS SÃO VÁLIDOS ATÉ 30/11/85.
PAGÁVEL EM QUALQUER AG. DO BANCO BRADESCO.

ASSINANTE	
ENDEREÇO	
BAIRRO	
CIDADE	**CEP**
ESTADO	**FONE**

CREDITAR BRADESCO — AG. CONSOLAÇÃO — C/C. Nº 73.966-9

VÁLIDO SE AUTENTICADO MECANICAMENTE PELO BANCO

VIA MICROHOBBY

PEÇA OS NÚMEROS ATRASADOS E COMPLETE A SUA COLEÇÃO

SIM, desejo receber os exemplares assinalados ao lado pelo preço de Cr$ 7.800,00 cada.

NOME	
ENDEREÇO	
BAIRRO	CEP
CIDADE	
ESTADO	FONE

☐ Nº 2
☐ Nº 4
☐ Nº 9
☐ Nº 10
☐ Nº 11
☐ Nº 12
☐ Nº 13
☐ Nº 14
☐ Nº 15
☐ Nº 16
☐ Nº 17
☐ Nº 18
☐ Nº 19
☐ Nº 20
☐ Nº 21
☐ Nº 22

TOTAL DO PEDIDO CR$ ____________________

ENVIO CHEQUE NOMINAL CRUZADO OU VALE POSTAL À
MICRODIGITAL ELETRÔNICA LTDA./ MICROHOBBY
CAIXA POSTAL 54.096 — PABX 255-0366 — CEP 01228 — SÃO PAULO
CHEQUE Nº BANCO
☐ VALE POSTAL